

# ADA USER JOURNAL

Volume 22  
Number 3  
September 2001

---

## Contents

	<i>page</i>
Editorial Policy for <i>Ada User Journal</i>	130
Editorial	131
News	133
Conference Calendar	171
Forthcoming Events	177
Articles	
Alan Burns	
“ <i>Guide for the use of the Ada Ravenscar Profile in high integrity systems – the work of the HRG</i> ”	182
Ada-Europe Associate Members (National Ada Organizations)	189
Ada UK 2001 Sponsors	190

# Editorial Policy for *Ada User Journal*

## Publication

*Ada User Journal* – The Journal for the international Ada Community – is published jointly by Ada Language UK Ltd and 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 Language UK Ltd and Ada-Europe an unlimited licence 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 Language UK Ltd, Ada-Europe or their 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. Electronic submission is preferred – typed manuscripts will only be accepted by the Editor 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 at:

[www.adauk.org.uk](http://www.adauk.org.uk)

There is no limitation on the length of papers, though a paper longer than 10,000 words would be regarded as exceptional.

# Editorial

The technical article in this issue provides an insight into the work of the Annex H Rapporteur Group (HRG). The HRG is concerned with the use of Ada in high integrity systems, having recently produced guidelines on the use of Ada in such systems. The article details the motivation and definition of Ravenscar, a safe tasking subset of Ada, supported by compiler vendors (notably Aonix) with minimal run-times.

The news section contains many interesting items. It is encouraging to see Ada success stories, notably the use of Ada in applications as diverse as set-top boxes and helicopters. Continuing application successes of Ada require good support by development tools – it is clear from the news that many Ada-related tools are still being supported and improved, including compilation environments.

The Forthcoming Events section gives details of the Symposium on Reliable Object-Oriented Programming (SROOP) 2001 (October 24<sup>th</sup>), being run by Ada Language UK. The program should interest all actively using, or considering the use of, the object-oriented features of Ada. Topically, one of the papers considers the extension of the Ravenscar profile to include limited object-oriented features.

Finally, in the previous issue (vol. 22, no. 2) Ian Gilchrist was named as the author of the Ada market report. Ian has asked me to credit Martyn Jordan for the hard work.

*Neil Audsley  
York  
September 2001  
Email: Neil.Audsley@cs.york.ac.uk*

# News

*Dirk Craeynest (ed)*

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

---

## Contents

	<i>page</i>
Ada-related Organizations	133
Ada Semantic Interface Specification (ASIS)	133
Ada and Education	133
Ada-related Resources	136
Ada-related Tools	136
Ada-related Products	144
Ada and Linux	148
Ada and Microsoft	148
References to Publications	150
Ada Inside	151
Ada in Context	157

---

## Ada-related Organizations

### 2001 SIGAda Election Results

*From: Hal Hart <Hal.Hart@acm.org>*  
*Date: Mon, 25 Jun 2001 18:12:04 -0700*  
*Organization: TRW*  
*Subject: 2001 SIGAda Election Results*  
*To: team-ada@acm.org*

SIGAda Members and Other Friends of SIGAda:

I am pleased to announce a successful SIGAda election for the two-year terms of officerships starting July 1. The winners are announced in the forwarded message below. In addition to the six elected officers, current SIGAda Chair Ben Brosgol will become the Past Chair, the 7th voting member of the SIGAda Executive Committee. The membership has chosen significant new blood on the new EC with a minority, only the new Chair and new Treasurer, having served as SIGAda officers previously (along with Ben). Nevertheless, all the winners have also been visible figures in the Ada community for many years, and most have made significant contributions to SIGAda [...]

On a personal note, it is with some fondness and some regret that I am now departing the SIGAda executive committee after more than 20 straight years -- continuously since the start of SIGAda's predecessor AdaTEC under SIGPLAN in 1981, before we "grew up" to become a full ACM SIG in 1984. [...]

Until next time (at the SIGAda 2001 conference Sep.30 - Oct. 4 in Minneapolis? -- See

<http://www.acm.org/sigada/conf/sigada2001/>),

Hal Hart, SIGAda Nominations Chair & Past Chair

[Extracted from a message from ACM HQ: -- dc]

The following individuals were elected to serve as SIGAda officers for the term July 1, 2001 - June 30, 2003:

Chair: Currie Colket The Mitre Corporation colket@mitre.org colket@colket.org

Vice Chair for Meetings and Conferences: David F. Harrison Logicon Technology Solutions dharrison@acm.org

Vice Chair for Liaison: Ann Brandon abrandon@sover.net

Secretary: Clyde Roby Institute for Defense Analyses roby@ida.org

Treasurer: John W. McCormick Dept. of Computer Science University of Northern Iowa mccormick@cs.uni.edu

International Representative: Jean-Pierre Rosen Adalog rosen.adalog@wanadoo.fr

---

## Ada Semantic Interface Specification (ASIS)

### ASIS Expression Resolver 2.0

*From: "Marc A. Criley" <mcqada@earthlink.net>*  
*Date: Thu, 09 Aug 2001 00:19:21 GMT*  
*Organization: Quadrus Corporation*  
*Subject: ANNOUNCE: ASIS Expression Resolver 2.0*  
*Newsgroups: comp.lang.ada*

ASIS Expression Resolver 2.0 is now available for download from AdaPower at <http://www.adapower.com/reuse/asisexp.html>. It has been tested with ASIS-for-GNAT 3.13p running on Linux.

ASIS\_Expression\_Resolver calculates the numeric value of an ASIS expression (i.e., where the Element\_Kind() is An\_Expression) if one can be semantically determined. Integer and floating point literals, named numbers, initialization expressions, First and Last values, and mathematical expressions can all be processed. Enumeration and character expressions can be evaluated in terms of their components, with the Pos,

Val, Pred, and Succ attributes all being properly handled. This package can be employed in the calculation of array bounds, range constraints, and wherever an expression is used whose value can be calculated solely through semantic analysis.

---

## Ada and Education

### Generics and Software Reuse

*From: Jack Beidler <beidler@cs.scranton.edu>*  
*Date: Wed, 6 Jun 2001 15:17:08 -0400*  
*Organization: Computing Sciences, University of Scranton*  
*Subject: Re: generics in Java*  
*To: team-ada@acm.org*

Terry Westley <twestley@acm.org> wrote:

> Some people should read the literature for design of other languages. Or, perhaps they just need to escape the NIH (not invented here) syndrome...

[and then quotes from <http://www.zdnet.com/zdnn/stories/news/0,4586,2769630,00.html>]

> Generics is controversial and has been widely debated inside and outside of Sun. Gosling said the concept is so complicated that no two developers could agree on exactly how it should be implemented. [...]

[See also "Generics in Java" further in this AUJ issue. -- dc]

It is a shame [those] in charge of Java [didn't] look at the generic/template models of Ada and C++. I am currently working on a couple of papers with two of my colleagues on Java's lack of template/generic capabilities. One of my colleagues is a C++ guy and we have been comparing Ada generics and C++ templates with an eye towards how we might like to see a generic/template feature implemented in Java.

The issue is much deeper than that. How many students go through an undergraduate program with any real software reuse experience (I can see it now, I am about to get flamed). People seem to claim as software reuse experience by just throwing any crap they want into a library and presto they are doing reuse.

Granted, there are some pretty good libraries out there in C++ and Java, but I just did a survey of CS 2 and CS 7 (Advanced data structures and algorithms) texts using C++. Would you believe that I could not find one book that uses C++ templates to encapsulate the basic data structures? If one of you know one, let me know.

Now keep in mind that just about all the Java book for these courses are "global replaces" of the various author's C++ books and you can see that we are facing a generation of programmers who have been raised on C++ and Java and have no real clue as to how to use a generic/template for the construction of reusable software!

If the software industry is in trouble now, I have a pretty good clue where it is going to be in the future as more and more schools dumb down there CS programs.

John (Jack) Beidler, Ph.D., Professor of Computer Science, Computing Sciences Department, University of Scranton, Scranton, PA 18510 Voice: (570)941-7774, Fax: (570)941-4250 Email: beidler@cs.Scranton.edu, Web: <http://www.cs.Scranton.edu/~beidler>

## What to Teach and in What Order?

From: Jacob Sparre Andersen  
<[sparre@nbi.dk](mailto:sparre@nbi.dk)>

Date: Thu, 14 Jun 2001 13:50:18 +0200

Subject: Discussion of introductory programming language

To: [team-ada@acm.org](mailto:team-ada@acm.org)

At slashdot.org there is currently a discussion of which language should be used for an introductory course in programming:

<http://slashdot.org/article.pl?sid=01/06/11/2021220&mode=nested>

I presume that some of the educators on this list might want to put in a word or two...

From: Michael Feldman

<[mfeldman@seas.gwu.edu](mailto:mfeldman@seas.gwu.edu)>

Date: Fri, 15 Jun 2001 12:49:56 -0400

Subject: Re: Discussion of introductory programming language

To: [team-ada@acm.org](mailto:team-ada@acm.org)

> [...] the choice of language should follow from educational goals, and I didn't see much about what the goals of CS1 should be in that thread. Perhaps Mike Feldman can enlighten us all ;)

Well, since [you] invited me to jump in...

CS education has always been fraught with debate over what to teach, and - more to the point - in what order to teach it. After 26 years of teaching courses ranging from freshman level to doctoral level and just about everything in

between, and nearly as many years of going to SIGCSE conferences and suchlike, I observe that:

- there is reasonably good consensus - in the respectable departments - on what we should teach over 4 years.

- there is no good consensus - and, IMHO, unlikely to be much in the foreseeable future - on the proper order.

The main difference between full-scale undergraduate programs, on one hand, and "training" programs like commercial certifications, 2-year community-college things, etc., on the other hand, is that the former focuses more on fundamentals - those things that don't change from year to year, while the latter focuses more on short-term concerns - products, "technologies", and skills.

Obviously there is a lot of "skill" stuff mixed into an undergrad program, but there is much more than that also.

If you're interested in reading about the nearest approximation to a consensus view of a good undergrad structure, see the accreditation standards published by CSAB, which is a creature of ACM and the IEEE Computer Society. Visit [www.csab.org](http://www.csab.org) for this. Click on "Criteria 2000" for details; click on "Comp. Sci. Profession" for the underlying philosophical basis.

You'll note that while some subjects in the standards are "obviously" introductory, and others are "obviously" advanced, in most cases CSAB does not prescribe an order to introduction.

There is little - if any - credible comparative research on the various tradeoffs in selecting presentation orders. For example, should we do UI's "early" [...]? Some do it, and claim success. Others present UI's "late", and claim success. Nobody can prove one is better than the other. Similar with O-O concepts, computer architecture, blah-blah.

In fact, if over 4 years, we are all teaching the subjects we agree on (and, as I said, there is pretty good agreement here), then the order really doesn't matter much and we can let each department follow its own tastes. I think this is the case, and probably always will be. It's not really productive to spend too much time trying to change some department's presentation order.

My bottom line to students AND to my colleagues in industry is this:

As a designer of a 4-year curriculum (and I AM the curriculum chair in my department!) I am obliged to produce *seniors* who have these basic attributes (in no particular order):

- are prepared for their first job, but also for their 10th job

- understand what changes and what doesn't

- can make design choices

- can tell the difference between revolutionary breakthroughs and marketing bullshit

- have enough math and science in their lives to be able to communicate with scientists and engineers

- can prepare a decent written report

- can give a decent oral presentation

- can work as a member of a team, not just as an isolated geek

(Aside: If they can read an API like Swing, fine, but if they can't, they can learn, because in college, they've learned how to learn. If they haven't learned how to learn, don't hire them!)

OK, that's what I'm obliged to produce. I am NOT (NOT!!!) obliged to produce:

- A worker bee who can jump right into a project with absolutely no on-the-job education into the tools and cultures of his/her employer. Employers must be willing to invest in their employees. If your employer won't invest in you, don't work there!

- A second-year summer intern who has exactly the right skillset the employer demands. You're an intern - there to learn, not to produce. If your employer doesn't understand this, quit and do something fun in the summer. I was a summer camp counselor till I got tired of it.

The above is part of my standard lecture to undergrad advisees, of whom I have roughly 125 right now.:-)

Whew! I'm sure that's more than you bargained for, but that's what you get for inviting a prof into this discussion!

Oh - if you have a few minutes, read a piece of mine:

[www.seas.gwu.edu/~mfeldman/papers/asspirations.html](http://www.seas.gwu.edu/~mfeldman/papers/asspirations.html)

Sigh... you asked for it.:-)

From: Michael Feldman

<[mfeldman@seas.gwu.edu](mailto:mfeldman@seas.gwu.edu)>

Date: Fri, 15 Jun 2001 13:30:21 -0400

Subject: Re: Discussion of introductory programming language

To: [team-ada@acm.org](mailto:team-ada@acm.org)

> Is the CSAB curriculum completely separate from the IEEE/ACM Computing Curricula 2001 (<http://computer.org/education/cc2001/ironman/cc2001/index.html>)? Do these organizations at least talk to each other?

(another full and lengthy answer...)

Yes, they talk to each other - CSAB is sponsored by ACM and IEEE. But their purposes and approaches are different.

The ACM/IEEE curriculum task forces (this is at least the 3rd over the years) produce recommended curriculum content in great detail, which usually has a certain influence in faculty circles, book publishers, etc. These are merely recommendations; they have no "legal" standing for accreditation, certification, or whatever. Also, they are purely \_curricular\_ recommendations; they do not speak to other issues like faculty qualifications, etc.

CSAB, on the other hand, is the designated agency that \_accredits\_ undergraduate CS programs. It is certified by the US Department of Education as an accrediting agency. This is similar to what AMA does in medicine, ABA in law, ABET in engineering, AIA in architecture, etc.

[aside]

For you non-US readers: in the US, the central government mostly stays out of education issues. The Feds supply only 7% of education funding here. For example, our "education ministry" does not accredit any ordinary education programs at any level; it only accredits the accreditation agencies!

Higher-education (postsecondary) accreditation is done two ways:

- colleges and universities are accredited by regional associations (GW is accredited by the Middle States Association of Colleges) and - in some States, not all - by the State government
- specific programs (typically only the professional ones) are accredited by the professional societies in those fields.

[end aside; back to CSAB]

The curriculum is a part of the accreditation process, but there are \_many\_ other factors such as stability, administrative commitment, lab resources, faculty qualifications, etc.

The CSAB curriculum standards are far more general, and any specific curriculum that meets the general standards is creditable.

Any reasonable implementation of ACM/IEEE curriculum recommendations would likely meet the CSAB curriculum standards, if it includes enough CS credit hours (40 minimum) to qualify.

Accreditation is "pass/fail" - there is no ranking. If you meet the standards, you pass. If you don't, you fail. In some sense, it puts a \_floor\_ under the definition of "respectable CS program." [...]

*From: "W. Wesley Groleau x4923"*  
*<wwgrol@ftw.rsc.raytheon.com>*  
*Date: Fri, 15 Jun 2001 13:52:08 -0500*  
*Subject: Re: Discussion of introductory programming language*  
*To: team-ada@acm.org*

> CS education has always been fraught with debate over what to teach, and - more to the point - in what order to teach it.

How many educators does it take to change a light bulb?

One to turn the bulb, but at least two others to offer unsolicited and mutually contradictory critiques of the methodology,

The original said "language teachers" (Earl W. Stevick, *Memory, Meaning, and Method*, 2nd Ed., 1996) but it applies pretty well to education--and software engineering--in general, doesn't it?

Wes Groleau,  
<http://freepages.rootsweb.com/~wgroleau>

## Ada in Malaysia

*From: byhoe@greenlime.com (Adrian Hoe)*  
*Date: 6 Jul 2001 18:57:08 -0700*  
*Subject: Re: is ada dead?*  
*Newsgroups: comp.lang.ada*

[...] One of UTM's (University of Technology Malaysia) KL campus was teaching and promoting Ada with a lot of confidence couple years ago when they were a joint-venture with Thomson CSF. Recently, I found that they had deserted Ada and switched to Java for the reason that there is no Ada market in Malaysia. Another reason came from one of the senior lecturers was that Ada was too old. I told the senior lecturer I could not believe what he was saying because they were so confident about Ada.

So, is Ada dead in Malaysia? I don't know how many Malaysians have joined CLA [= the comp.lang.ada newsgroup --dc], but I will say that Ada is not dead in Malaysia.

Reason? UTM's main campus in JB is teaching Ada in general and real time programming and there are as many as 120 students right now, yes, today!

Lexical Integration (M) Sdn Bhd, the company I work with, although not as aggressive as 4-5 years ago, is still promoting Ada. Our R&D division uses Ada for research projects. 100% of all works in Lexical are using Ada. Lexical Integration will emerge as Ada authority in Malaysia in no time to come and that's our ultimate goal!

Ada is a programming language appreciated by engineers who know the benefits. Java is a programming language appreciated by people who likes to read ads and listen to marketing persuasion. I do not intend to flame Java. It is a language with its own benefits and strength. This is what actually happened in Malaysia. People like to follow the newest trends.

In [some] universities (Malaysian, OK?), programming languages are taught not because of teaching the students of

programming concepts, but for the sake of market requirement. That's the most pathetic and irresponsible decision.

## Ada Training

[This information is included as examples of public Ada training courses: many are being organized regularly. For more, see also "Ada Training" in AUJ 21.3 (October 2000), p.161, and check out "Ada Education and Training" at <http://www.cs.kuleuven.ac.be/~dirk/ada-belgium/training/> on the Ada-Belgium web-server. -- dc]

*From: "Ed Colbert"*  
*<colbert@abssw.com>*  
*Date: Fri, 20 Jul 2001 14:58:45 -0700*  
*Subject: Announcing Public Ada 95 & Ada 83 Classes*  
*Newsgroups: comp.lang.ada*

Absolute Software will be holding a public Ada 95 on the week of 20 August and a public Ada 83 class on the week of 15 October. Both will be held in Carlsbad, CA. You can find a full description and registration form on our web-site, [www.abssw.com](http://www.abssw.com). Click the Public Courses button in the left margin. (We also offer classes on object-oriented methods and other object-oriented languages.)

If there is anything you'd like to discuss, please call, write, or send me E-mail.

*From: rod@praxis-cs.co.uk (Rod Chapman)*  
*Date: 16 Jul 2001 03:27:33 -0700*  
*Subject: ANNOUNCE: SPARK Training Course, September, UK.*  
*Newsgroups: comp.lang.ada*

We're pleased to announce the second public "High Integrity Software Engineering with SPARK" course for this year.

This is a 4-day course for managers, regulators and engineers, which presents the principles of the development of high integrity software, and the related certification requirements. It explains the rationale of SPARK, outlines the language and the principles of static code analysis, and presents the role of the SPARK Examiner in systematic program development.

When: 4th - 7th September 2001 Where: Praxis Critical Systems' offices in Bath, UK.

For more information, please contact Fiona Joy, Praxis Critical Systems Limited, 20 Manvers Street, Bath BA1 1PX, UK, email: [sparkinfo@praxis-cs.co.uk](mailto:sparkinfo@praxis-cs.co.uk), phone: +44 (0)1225 466991, fax: +44 (0)1225 469006

*From: rod@praxis-cs.co.uk (Rod Chapman)*  
*Date: 26 Jul 2001 08:04:23 -0700*  
*Subject: ANNOUNCE: USA SPARK Training course!*

*Newsgroups: comp.lang.ada*

We're pleased to announce the first public "Software Engineering with SPARK" course in the USA.

This course is being run in collaboration with and will be hosted by the School of Engineering and Computer Science at Oakland University, Rochester, Michigan.

The course will run from 24th to 28th September 2001.

Full details, and a booking form at available in PDF at [www.sparkada.com](http://www.sparkada.com) under "Training". Alternatively, drop us a line at [sparkinfo@praxis-cs.co.uk](mailto:sparkinfo@praxis-cs.co.uk)

Rod Chapman, SPARK Team, Praxis Critical Systems

About SPARK

SPARK is a high level programming language developed specifically to support high integrity applications where correct operation is vital either for reasons of safety, security or business integrity. Writing code in SPARK automatically eliminates the possibility of introducing many types of programming error.

SPARK is a cost-effective engineering solution. It meets the customer's needs by providing a high quality product; the developer's needs by reducing cost and risk; and the regulator's needs by simplifying the production of the required evidence of fitness for purpose. SPARK is the logical choice.

## Ada in France

*From: Pierre Habraken*

*<Pierre.Habraken@imag.fr>*

*Date: Tue, 21 Aug 2001 22:32:28 +0200*

*Organization: Universite Joseph Fourier*

*Subject: Abort statement and Linux*

*To: "GNAT Discussion List"*

*<gnatlist@lyris.seas.gwu.edu>*

[...] we (at Grenoble University, France) plan to use basic Ada multi-tasking (pure rendez-vous, no shared memory) in lab work on advanced algorithmics at undergraduate level (please be indulgent for my poor English...). Most of our lab rooms will be equipped with Linux machines (one for two students) in a near future. [...]

IMAG-LSR, BP72, 38402 Saint Martin D'Heres Cedex Tél: 04 76 82 72 48, Fax: 04 76 82 72 87

## Ada-related Resources

### Ada 95 BNF

*From: Michael Feldman*

*<mfeldman@seas.gwu.edu>*

*Date: Wed, 13 Jun 2001 15:31:04 -0400*

*Subject: Re: Ada 83/95 Grammar B.N.F.*

*To: team-ada@acm.org*

> Could I please be pointer to a website and/or a good PDF file that explains in details the Ada 83 and 95 Grammar in B.N.F. format?

See <http://cui.unige.ch/db-research/Enseignement/analyseinfo/Ada95/BNFindex.html> for a nice text and graphic rendering of Ada 95 BNF.

They don't have an Ada 83 one there, but syntax-wise there are few differences. General access types, protected types, and tagged types are the ones that come to mind. The syntactic leap is `_much_` smaller than from C to C++, for example.

## GNAT-related Web Pages & Email Lists

*From: "Gautier de Montmollin"*

*<gdemont@hotmail.com>*

*Date: Wed, 04 Jul 2001 22:13:08 -0000*

*Subject: GNATList page has moved*

*To: "GNAT Discussion List"*

*<gnatlist@lyris.seas.gwu.edu>*

Hi - just a word to say that the "GNATList page" has moved from NBCi webspace to

<http://www.diax.ch/users/gdm/gnatlist.htm> as well as the "GNAT/DOS" page : <http://www.diax.ch/users/gdm/gnatdos.htm>

*From: Ted Dennison*

*<dennison@telepath.com>*

*Date: Mon, 06 Aug 2001 14:28:23 GMT*

*Subject: Re: gnat newsgroups*

*Newsgroups: comp.lang.ada*

There are email lists available [...].

Gnatlist is at [http://lyris.seas.gwu.edu/cgi-bin/lyris.pl?enter=gnatlist&text\\_mode=0](http://lyris.seas.gwu.edu/cgi-bin/lyris.pl?enter=gnatlist&text_mode=0) . [...]

T.E.D., <http://www.telepath.com/dennison/Ted/TED.html>

## Comments on the Ada Standard

*From: "Marc A. Criley"*

*<marccriley@earthlink.net>*

*Date: Thu, 23 Aug 2001 07:24:36 -0400*

*Subject: Re: Proposal: For-Loop Iteration for Real Type*

*To: team-ada@acm.org*

Just as a point of information:

There is an open mailing list for discussion of new, and changes to, Ada language features. See the Introduction to the Ada 95 Reference Manual, "Instructions for Comment Submission", but instead of the email address given there ([ada-comment@sw-eng.falls-church.va.us](mailto:ada-comment@sw-eng.falls-church.va.us)), use [ada-comment@ada-auth.org](mailto:ada-comment@ada-auth.org).

[See also "Maintenance of Ada Standard" in AUJ 21.2 (July 2000), p.98. -- dc]

Marc A. Criley, Senior Staff Engineer, Quadrus Corporation, [www.quadruscorp.com](http://www.quadruscorp.com)

[Randy Brukaradt

*<Randy@rrsoftware.com> replied: -- dc]*

Well, if you're using the consolidated Ada 95 Reference Manual (with Technical Corrigendum 1), you'll find that those instructions have been updated. And if you're not, why not?? :-) See <http://www.ada-auth.org/arm.html>.

[See also "Revised Ada Reference Manual available" in AUJ 21.4 (January 2001), p.238. -- dc]

Printed copies of the consolidated Ada 95 Reference Manual are going to be available soon. Watch this space...

## Ada-related Tools

### Hash Functions

*From: James Rogers*

*<jimmaureenrogers@worldnet.att.net>*

*Date: Mon, 28 May 2001 00:23:07 GMT*

*Subject: Re: function hash*

*Newsgroups: comp.lang.ada*

> I am a student of Computer Science and I am looking for an easy hash function that can work with 15 or more characters... It's for working in a table of 500-1000 elements. ANY idea?

You might find the package GNAT.HTable to be useful. This package is included in the public GNAT distribution.

Jim Rogers, Colorado Springs, Colorado, USA

*From: Jeffrey Carter <jrcarter@acm.org>*

*Date: Mon, 28 May 2001 06:47:55 GMT*

*Subject: Re: function hash*

*Newsgroups: comp.lang.ada*

You might find the article "Fast Hashing of Variable-Length Text Strings" by P. K. Pearson (CACM 1990 Jun) of interest. It describes a fast "hashing function specifically tailored to variable-length text strings".

An implementation of this function is part of the PragmAda Reusable Components, available from

<http://home.earthlink.net/~jrcarter010/pragmarc.htm>

or from the mirror at [www.adapower.com](http://www.adapower.com).

[See also "Secure hash functions" in AUJ 20.1 (April 1999), pp.20-21. -- dc]

## Pure Variable\_String Package

*From: "Jean-Pierre Rosen"*

*<rosen@adalog.fr>*

*Date: Thu, 7 Jun 2001 10:07:51 +0200*

*Organization: Adalog*

*Subject: Re: Some problems with string type (Exact String?)*  
*Newsgroups: comp.lang.ada*

[...] Alternatively, you can use the Variable\_String package that is available on Adalog's components page:  
<http://www.adalog.fr/compo2.htm>

Note that this package is pure, which (unlike the other packages) makes it usable for distributed applications.

J-P. Rosen, [rosen@adalog.fr](mailto:rosen@adalog.fr),  
<http://www.adalog.fr>

## Ada Structured Library (AdaSL)

*From: Corey Minyard*  
*<minyard@acm.org>*  
*Date: Sat, 16 Jun 2001 14:06:57 GMT*  
*Subject: Ada Structured Library (AdaSL, or ASL) on SourceForge*  
*Newsgroups: comp.lang.ada*

I have put AdaSL on SourceForge for a more public presence. I am not currently working on it because I am not using Ada right now :-), but I'd like it to be in a visible place where others can work on it and participate.

So visit <http://adasl.sourceforge.net> if you are interested.

[See also "Ada Structured Library (ASL)" in AUJ 20.1 (April 1999), p.15, and in AUJ 19.3 (October 1998), pp.149-150, for an overview. -- dc]

## Dual\_IO Package

*From: "Gautier Write-only-address"*  
*<gautier\_niouzes@hotmail.com>*  
*Date: Thu, 26 Jul 2001 13:09:25 +0000*  
*Subject: Dual\_IO (2)*  
*Newsgroups: comp.lang.ada*

Hello - just a word to say that Dual\_IO is now made - i.e. a package that outputs all I/O from/to Standard also to a log file. It is in paqs.zip @ URL below.

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

NB: Do not answer to sender address, visit the Web site!

## Task-safe Debug Package

*From: "Jean-Pierre Rosen"*  
*<rosen@adalog.fr>*  
*Date: Tue, 21 Aug 2001 09:40:52 +0200*  
*Organization: Adalog*  
*Subject: Re: Abort statement and Linux To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

> In my time I've found a \*lot\* of tasking errors caused by unprotected use of Text\_IO. [...] But I'd say it most definitely \*is\* bad practice. The last thing you want to do with debugging code is insert the possibility of having seemingly random errors.

Perfect opportunity for a shameless plug

If you want to debug multi-task programs, have a look at package Debug on Adalog's components page (<http://www.adalog.fr/compo2.htm>). It is perfectly task-proof, and even allows tracing protected calls (i.e. it provides trace functions that are NOT potentially blocking).

## Examples of How to Use Streams

*From: James Rogers*  
*<jimmaureenrogers@worldnet.att.net>*  
*Date: Fri, 25 May 2001 22:00:27 GMT*  
*Subject: Re: Use of streams with Ada95*  
*Newsgroups: comp.lang.ada*

> Any one got an example of using streams for offline storage, i.e. to provide interface to write and read back from storage. Booch and Barnes don't seem to describe this in much detail

Cohen (Ada as a Second Language, Second edition) describes this in adequate detail including examples. His section on streams starts on page 786 and continues through page 794.

*From: "Marin David Condic"*  
*<marin.condic@pacemicro.com>*  
*Date: Tue, 29 May 2001 10:28:47 -0400*  
*Subject: Re: Use of streams with Ada95*  
*Newsgroups: comp.lang.ada*

Look at my web page (<http://www.mcondic.com/>) on the Ada Programming page for a collection of code named "gnat\_examples.chop" Search it for "Streams". There are a couple of programs in there that do I/O to files using streams. Note that to utilize streams with structured data (records, etc), you have a kind of two stage process. First you have to use the Read and Write (or Input & Output) to get the structured data into the stream, then use the Stream\_IO facilities to get the data out or in to the file.

I may have a more detailed example around here somewhere of actually making that happen if you need it, but I'd have to dig. If you know how to get the data into and out of the stream, the I/O should be pretty straightforward.

Marin David Condic, Senior Software Engineer, Pace Micro Technology Americas, [www.pacemicro.com](http://www.pacemicro.com), e-Mail: [marin.condic@pacemicro.com](mailto:marin.condic@pacemicro.com), Web: <http://www.mcondic.com/>

## FSMedit 1.4 - Editor for Finite State Machines

*From: Christoph Grein*  
*<christoph.grein@eurocopter.com>*  
*Date: Thu, 31 May 2001 12:05:15*  
*Subject: Announcement: Finite State Editor*  
*Newsgroups: comp.lang.ada*

FSMedit V1.4 has been released. This is the first full implementation.

The licensing information has been updated.

FSMedit is an editor for finite state machines. It is written in Ada with the graphical user's interface by Claw. For token recognition, OpenToken is used.

It is released under GPL and comes with full documentation.

Download from <http://home.T-Online.de/home/Christ-Usch.Grein/Ada/> or from AdaPower.

[See also AUJ 22.1 (April 2001), p.10. -- dc]

## AdaBill - Billing System for Small Companies

*From: Adrian Knoth*  
*<adi@drcomp.erfurt.thur.de>*  
*Date: 31 May 2001 19:25:03 GMT*  
*Organization: Modern Electronics*  
*Subject: TCP/IP primitives package for Ada95*  
*Newsgroups: comp.lang.ada*

[...] And now some advertisement for my own project: AdaBill (a billing system for small companies) is more or less ready to use, but I plan to expand it. It seems that I'll have some time within the next months, but I would be glad to have a co-developer. There is also the need to implement the web-client in PHP.

So if someone is interested you may have a look at [adi.thur.de/adabill.html](http://adi.thur.de/adabill.html)

## GVD 1.2.0 - GNU Visual Debugger

*From: Arnaud Charlet*  
*<charlet@gnat.com>*  
*Date: Wed, 13 Jun 2001 14:09:28 GMT*  
*Subject: ANNOUNCE: GVD 1.2.0*  
*Newsgroups: comp.lang.ada*

We are pleased to announce the availability of GVD, the GNU Visual Debugger, a general purpose graphical debugger front-end licensed under the GNU General Public License.

[See "GNU Visual Debugger (GVD 1.1.0)" in AUJ 22.2 (June 2001), p.73, for a full announcement. -- dc]

You can download GVD (sources and binaries for GNU/Linux x86, Solaris sparc and x86, Windows NT/2000, DEC Unix/Tru64, HP-UX, UnixWare, IRIX, AiX) and get more information at <http://libre.act-europe.fr>

New features in GVD 1.2.0:

- Under Windows, the special pathnames //cygdrive/x/ and //x/ are now recognized and converted into appropriate names.



- The File->New Debugger dialog now provides the ability to replace the current debugger by default.
- The "run" and "start" buttons now set automatically the name of the module under VxWorks.
- The File->Attach Process menu now lists the tasks on the target when using Tornado VxWorks.
- The --target option now allows multiple ':' characters so that you can specify machine:1234 as the machine (& port number). See the documentation for more details.
- The file corresponding to the main procedure is now displayed when loading an Ada executable (rather than the procedure containing the elaboration calls).
- The data window is now zoomable
- The breakpoint editor has been enhanced, and can now set advanced properties like ignore\_count, commands to execute when reaching the breakpoint, or conditions to activate the breakpoint
- The handling of commands has been improved and is more reliable. Caching of commands has also been put in place giving better user feedback on slow machines.
- New item in the source editor contextual menu: display dereferenced variable
- New item in the source editor contextual menu: break on subprogram
- The option "break on exceptions" has been implemented
- It is possible to delete an exception by pressing the 'Delete' key in the Breakpoints Window.
- The Tab key is now supported in the debugger window.

If you are interested in participating in the GVD development, do not hesitate to contact us (mailto:gvd-devel@lists.act-europe.fr)

Arnaud Charlet <charlet@act-europe.fr>

## AdaVox 0.5 - Wave Sound File Player

*From: "Warren W. Gay VE3WWG"*  
 <ve3wwg@home.com>

*Date: Mon, 02 Jul 2001 01:27:26 -0400*

*Subject: AdaVox-0.5 Released*

*To: "GNAT Discussion List"*

<gnatlist@lyris.seas.gwu.edu>

AdaVox 0.5 is a wave player, which is written almost entirely in Ada95, using GNAT (3.13p). It has been tested for FreeBSD and Linux (Red Hat 7.0, 2.2.16 kernel). It currently does not record wave samples, but this will be coming in the future.

If you are running a big-endian machine like the DEC Alpha, this should also be able to work with wave files that many wave players cannot, on a big endian host. AdaVox adjusts its endian handling, based upon the host characteristics and the expected endianness of the sound file. This is untested, but I am interested to hear any success stories. mailto:ve3wwg@home.com if you have success.

Download: <http://members.home.net/ve3wwg/adavox-0.5.tar.gz>

Features:

Wave File Formats (\*.wav):

- Plays standard PCM wave files
- Plays Microsoft ADPCM wave files
- Plays IMA/DVI ADPCM wave files [1]
- Plays u-Law wave files
- Plays a-Law wave files

Sun file formats (\*.au):

- Plays standard PCM wave files
- Plays u-Law wave files
- Plays a-Law wave files

Other Features :

- Plays several files in sequence [2]
- Endian neutral \*\*
- Ada 95 sound packages provided
- Optional realtime scheduling [5]
- For Linux and BSD (FreeBSD tested)
- Netscape friendly

[1] see notes at the bottom

Limitations :

- Does not record sound files yet
- Will not play from a pipe or socket [6]

The project is also structured to permit the re-use of some of the WC.Streams.\* Ada packages. This will allow developers to re-use some of the provided software modules.

Notes:

[1] The CODEC has been designed to work for 2, 3, 4, and 5 bit samples. The 2 and 5 bit formats are not standard formats, but are designed to be compatible with CoolEdit-2000. The provided CODEC works with all CoolEdit-2000 generated monophonic files. However, CoolEdit-2000 generated stereo files do not seem to play correctly, unless the 4-bit format is chosen. Even though CoolEdit-2000 plays its own files correctly, it is believed that CoolEdit-2000 incorrectly interleaves the samples for stereo files.

[2] The files are played in the sequence given on the command line.

[3] There are some older mainframe CPU types that use very odd endian formats that cannot be supported. However, it is not likely that this will be an issue with modern equipment.

[4] The magic is localized in the source files wc-endian.ad[sb]. The rest of the application is totally unaware of the

endian issue, which makes programming life much easier!

[5] The adavox command will establish a default realtime priority of LO+1, where LO is the lowest realtime priority (on platforms that support it). This can be overridden on the command line with the -R option. For this to work, it must be installed setuid root (installed this way by default). Otherwise, the command runs at normal priorities.

[6] Many people have asked me to fix wavplay to play files from a pipe or socket. I seriously looked at providing this functionality in adavox, but concluded that it is just too much trouble for the benefit gained. Some files like Microsoft's \*.wav file are just too complex to deal with in a sequential fashion. Instead, the extra effort went into providing endian-neutral support. Look for server support to address streaming in the future.

Future development:

There was some MP3 support started, but this port is a large task. I have backed away from this at the present time, in favour of getting adavox out there for others to use. It is useful for a Netscape helper application, replacing wavplay. It is hoped that adavox will eventually retire the wavplay project (currently wavplay 1.4).

In the mean time, the next effort will be focused on record functionality. This will eventually include conversion of one wave file format to another (in addition to recording different formats).

MP3 support needed:

If someone wants to submit MP3 support (in Ada), I will gladly integrate it into the existing code base. However, before submitting code, please keep the design goal firmly in mind. There is a minimum of module interaction within the current design. In a top level form, the interfaces are:

```
[SRC-FORMAT] -> [CODEC] ->
[DEST-FORMAT]
```

The [SRC-FORMAT] represents one Ada stream (WC.Streams.Wave or WC.Streams.AU for sound files) as the source stream. The [DEST-FORMAT] is currently only supported by WC.Streams.DSP (the Digital Sound Processor). Eventually the WC.Streams.Wave and WC.Streams.AU packages will also serve as destinations, when the code is complete (allowing recording or file format conversions). Obviously, in order to record samples, the WC.Streams.DSP package will also have to be finished to act as a source.

In the middle of all of this is the [CODEC]. It simply pumps wave samples from the source to the destination, performing decompression as required. This looks conceptually simple,

but the implementation is very tricky in Ada without bringing in extra inter-module dependencies and APIs. MPEG support is nearly a nightmare, when you include ID3 Tag support. Maybe this is the challenge you have been waiting for?

OGG-VORBIS support:

It would also be nice to have Ada95 OGG-VORBIS support. Perhaps another ambitious person would like to have a go at it?

[For some context, see also "C developer switches to Ada for Linux development" in AUJ 21.4 (January 2001), pp.246-248. -- dc]

*From: "Warren W. Gay VE3WWG" <ve3wwg@home.com>  
Date: Tue, 03 Jul 2001 21:52:41 -0400  
Subject: AdaVox-0.51 Update  
To: "GNAT Discussion List" <gnatlist@lyris.seas.gwu.edu>*

It was pointed out by Wilhelm Spickermann (Wilhelm.Spickermann@t-online.de) that I was using an older version of the Booch components (thanks). If you have already downloaded and installed version 0.5, there is no reason to upgrade.

However, release 0.51 has been made available, if you want the more current sources (there was a small change to linked lists between the older and the bc-20010513 Booch Components). If you want to get the most recent sources, download them from :

<http://members.home.net/ve3wwg/adavox-0.51.tar.gz>

or simple check

<http://members.home.net/ve3wwg> for the latest info, and patches.

## Adump 1.0 - Backup Software

*From: David Brown <gnatlist@davidb.org>  
Date: Sun, 5 Aug 2001 17:27:30 -0700  
Subject: Announce - Adump 1.0  
To: "GNAT Discussion List" <gnatlist@lyris.seas.gwu.edu>*

I am pleased to announce the release of Adump 1.0. Please refer to my web page <<http://www.davidb.org/adump/>> for more information.

What is adump? Adump is intended to eventually be a full suite of backup software. At this point, it can do full and incremental backups, and do simple media management.

This release is primarily intended for other interested developers to look at, although I have been using it for about a year to backup my machine.

Best of all, Adump is entirely written in Ada95.

David Brown, [adump@davidb.org](mailto:adump@davidb.org)

[See also "Adump - Backup suite project" in AUJ 21.4 (January 2001), p.225-226. - - dc]

## GNAT for Mac OS X

*From: "Roy M. Bell" <rmbell@acm.org>  
Date: Sat, 25 Aug 2001 19:19:14 GMT  
Subject: GNAT for Mac OS X  
Newsgroups: comp.lang.ada*

Jim Hopper (hopperj@macconnect.com) has asked me to post the following:

There are a whole team of us working on this port (to Mac OS X) right now. Myself (i.e., Jim Hopper), alan and andrew reynolds, Stephen J. Bespalko, John Matthews, and others who i am probably forgetting off the top of my head. In addition Stan Shebs at Apple who is the keeper of the gcc 3.0 port is also working with us. We have also been talking to the Apple pthreads team. In addition the folks at ACT have supplied us with gnat sources and answers to some of our questions (those which were not Apple related)

We have had to do a number of things to the gcc port to make it support gnat. There were a few shortcuts taken in Apples port of gcc that have the backend calling the front end. And while this works fine for the c, and C++ front ends it was intended for and seems to work for fortran, it doesnt work at all for ada. So Stan, Alan, Andrew, and myself have made changes to gcc to try to fix some of these. We have fixed a few of them but we are not done yet. Seems each time we get past one of these another crops up.

We also had to add trampoline support to Apples gcc as it was never made to work on the OS X port for any language. So nested functions never worked in Apples gcc, but they will in 3.0 if you have the patch now. Stan is going to get it into the mainline sources. Right now we are working on another gcc problem. We succeeded in building a gnat1, but it won't bootstrap which we are working on.

Pthreads is another matter, Apples implementation of pthreads is not complete. However the Apple folks are willing to look at adding what we need and one of the team is corresponding with them about it. In the short term we have fsu threads ported so we will simply have to use that in the short term.

We have set up a public mailing list for those who want to be kept up to date with whats going on. Its at:

[http://www.lists.apple.com/cgi-bin/mwf/board\\_show.pl?id=3&pg=2](http://www.lists.apple.com/cgi-bin/mwf/board_show.pl?id=3&pg=2)

Apple is publicising it on there system and making it easy to get to but its being hosted with the kind support of George Washington University and Mike Feldman. We have a developers list as

well but we are trying to keep people off that unless they can help with some phase of this.

Once we get the basic compiler going we will have a lot of work on the bindings to carbon coco etc.

Its not near as neat as it would be if we had a gcc that was complete. But with the help of stan shebs and the reynolds brothers we are getting there.

## Random Number Generators

*From: Jeffrey Carter <jrcarter@acm.org>  
Date: Fri, 15 Jun 2001 03:06:12 GMT  
Subject: Ada 83 Random Generator - or ??  
Newsgroups: comp.lang.ada*

> Is there out there a "free" package vaguely similar to Ada.Numerics.Discrete\_Random, which is portable to all Ada 83 AND 95 compilers ?

The PragmAda reusable components includes an implementation of Marsaglia's "universal" random number generator, and I may have a copy of the 83 version around somewhere. In any case, there are not too many 95-isms in the 95 version, so you could clean it up yourself pretty easily. It includes a function that returns a uniformly distributed integer in a given range, which is "vaguely similar to Ada.Numerics.Discrete\_Random". It should also be completely portable once converted completely to Ada 83.

*From: gautier\_niouzes@hotmail.com (Gautier)*

*Date: 15 Jun 2001 04:59:34 -0700  
Subject: Ada 83 Random Generator - or ??  
Newsgroups: comp.lang.ada*

Check U\_Rand.\* in mathpaqs.zip @ <http://www.diix.ch/users/gdm/gsoft.htm>

## Arbitrarily Long Integers

*From: "Jesse Farmer" <tujuggernaut@gmx.net>  
Date: Mon, 25 Jun 2001 17:54:29 -0400  
Subject: very large number package  
Newsgroups: comp.lang.ada*

I was wondering if anyone knows of a good, free, package similar to the Number Theory Library available for C++? <http://www.shoup.net/ntl/>

Basically, I am looking for a package providing arbitrarily long integers, and matrices that can contain such integers. It would be nice if conversions from standard ints to these long ints, all basic operations, and operations that can be optimized (like matrix ^2) were provided. [...]

I really do appreciate the fact that so many experienced Ada programmers take the time to answer simple questions of new Ada programmers like me.

*From: "Gautier Write-only-address"*  
 <gautier\_niouzes@hotmail.com>  
*Date: Tue, 26 Jun 2001 08:50:24 -0000*  
*Subject: Re: very large number package*  
*Newsgroups: comp.lang.ada*

You'll find 5 free packages @  
<http://www.chez.com/bignumber/>. For  
 matrices, pick a generic matrix package  
 (\*), feed your large int type & ops. and  
 compile the whole with all optimization:  
 cross-unit inlining, loop unrolling etc.

(\*) a limited one on the page below, in  
[mathpaqs.zip](http://mathpaqs.zip)

Gautier,  
<http://www.diach.ch/users/gdm/gsoft.htm>

## On Graphics in Ada

*From: "M. A. Alves"*  
 <maa@borba.ncc.up.pt>  
*Date: Thu, 7 Jun 2001 16:31*  
*Subject: Re: Graphics*  
*Newsgroups: comp.lang.ada*

> How would I implement graphics using  
 Ada? I'm looking for simple  
 lines/points/colors ...

Here's a minimal package for that:  
[http://lexis.di.fct.unl.pt/ADaLIB/  
 vga\\_screen.htm](http://lexis.di.fct.unl.pt/ADaLIB/vga_screen.htm)

> ... and maybe a routine to display  
 BMP's or another file format.

It seems PBM (Portable Bit Format),  
 formerly NetPBM, is the way to go, at  
 least on Linux systems.

> Where can I find more information on  
 this?

On Google ;-)

[And in most issues of AUJ... -- dc]

Mario Amado Alves LIACC, room 221,  
 Rua Campo Alegre, 823, P-4150 Porto,  
 Portugal tel 351+226078830, ext 121, fax  
 351+226003654, mob 351+939354002

*From: tmoran@acm.org*  
*Date: Thu, 07 Jun 2001 19:44:42 GMT*  
*Subject: Re: Graphics*  
*Newsgroups: comp.lang.ada*

Look at [www.adapower.com](http://www.adapower.com). There are  
 thin and thick bindings to Windows  
 (mention of BMP suggests Windows or  
 DOS is the platform here) and SVGA IO  
 for DOS.

*From: "Joachim Schröer"*  
 <joachim.schroer@web.de>  
*Date: Thu, 7 Jun 2001 22:36:09 +0200*  
*Subject: Re: Graphics*  
*Newsgroups: comp.lang.ada*

If you look for e.g. BMP-display on  
 windows you could look into the code on  
 my page [www.adapower.com/schroer](http://www.adapower.com/schroer). In  
 the package [opengl\\_bitmaps](http://www.adapower.com/opengl_bitmaps) in [ogl-src.zip](http://www.adapower.com/opengl_src.zip)  
 you find source for reading and  
 displaying BMPs.

You could also look into my mail from []  
 May where the download of all files is  
 described [...]

[See "OpenGL Bindings and Demos" in  
 AUJ 22.2 (June 2001), pp.72-73. -- dc]

## DirectX

*From: "Mark Maratea"*  
 <mmaratea@pacemech.com>  
*Date: Tue, 14 Aug 2001 10:14:08 -0400*  
*Subject: Has anyone experimented with*  
*DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

I've noticed alot of OpenGL graphics  
 packages but I haven't been able to find a  
 single reference to DirectX. I could really  
 use some help as I'm attempted to  
 develop a DirectX graphics package for  
 Ada. [...]

*From: "Gautier de Montmollin"*  
 <gdemont@hotmail.com>  
*Date: Tue, 14 Aug 2001 15:39:18 +0000*  
*Subject: RE: Has anyone experimented with*  
*DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

> Excellent idea. Lot of people at my  
 company want to use Ada but because  
 it doesn't have binding to DirectX shy  
 away from it.

The binding is easy (\*) to obtain, through  
 GNATCOM. You spot the DLL for VB  
 (in system directory) that contains the  
 COM information. Run "bindcom  
 dx8vb.dll tralala" and it will spit a good  
 mega of Ada packages, structured with  
 children packages etc. and "tralala" as  
 parent name. That's all! (a name like DX8  
 will be more appropriated than tralala).

The tough thing is what to make with this  
 binding, DX seeming a masterpiece of  
 bad organization. Someone who knows  
 VB or VC++ or DX more than empty set  
 and can pick from tutorials on the Web  
 will do the job easily.

(\*) I mean, really easy, [...]

Gautier, [http://www.mysunrise.ch/  
 users/gdm/gsoft.htm](http://www.mysunrise.ch/users/gdm/gsoft.htm)

## AdaSDL - Binding to Simple DirectMedia Layer (SDL)

*From: Antonio M. F. Vargas*  
 <avargas@adapower.net>  
*Date: Sat, 28 Apr 2001 18:14:44 +0100*  
*Subject: AdaSDL in sourceforge*  
*Newsgroups: comp.lang.ada*

[...] I'm moving AdaSDL files to:  
[www.sourceforge.net/projects/adasdl](http://www.sourceforge.net/projects/adasdl)  
 Right now you can get some files via cvs.  
 [...]

mailto: antonio.vargas@clix.pt

[See AUJ 22.1 (April 2001), p.12, for  
 more information. -- dc]

*From: Frode Tenneboe*  
 <ft@edh.ericsson.se>  
*Date: Wed, 15 Aug 2001 08:48:*

*Subject: RE: Has anyone experimented with*  
*DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

> Lot of people at my company want to  
 use Ada but because it doesn't have  
 binding to DirectX shy away from it.

Why? One of the main points about Ada  
 is it's cross platform abilities. DirectX is  
 definitely NOT cross platform - just the  
 opposite. [...]

SDL (<http://www.libsdl.org/>) should  
 provide you with a cross platform  
 graphics and audio library.

Frode Tenneboe, Ericsson Radar AS,  
 Isebakkeveien 49, N-1788 Halden,  
 ft@edh.ericsson.se, phone +47 69 21 41  
 47

[From a later message: -- dc]

It also contains a pointer to a ready made  
 thin Ada binding.

[And on the remark that the binding  
 seems to be for an older version of SDL,  
 Eric Streit <Eric.Streit@wanadoo.fr>  
 replied: -- dc]

I tried the Ada binding with the last  
 versions of the SDL libraries, and it  
 works without any problem , and fast !

*From: "Gautier de Montmollin"*  
 <gdemont@hotmail.com>  
*Date: Mon, 27 Aug 2001 12:35:13 +0000*  
*Subject: RE: Has anyone experimented with*  
*DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

> Take a look at <http://www.libsdl.org/>. It  
 also contains a pointer to a ready made  
 thin Ada binding.

Interesting is that the Windows port uses  
 DirectX (according to the doc), so it  
 should make everybody happy!

I see following advantages:

- portable
- no need to change everything when DX  
 (N+1) is out
- Ada binding existing and tested

Gautier, [http://www.mysunrise.ch/  
 users/gdm/](http://www.mysunrise.ch/users/gdm/)

## GtkAda 1.2.12 Release

*From: Arnaud Charlet*  
 <charlet@gnat.com>  
*Date: Wed, 13 Jun 2001 14:06:14 GMT*  
*Subject: ANNOUNCE: GtkAda 1.2.12*  
*Newsgroups: comp.lang.ada*

We are happy to announce the  
 availability of GtkAda 1.2.12.

GtkAda is an Ada95 graphical toolkit  
 based on Gtk+ version >= 1.2.2. It allows  
 you to develop graphical applications in  
 Ada95 using Gtk+.

The primary download site is  
<http://libre.act-europe.fr/GtkAda>

New features in 1.2.12 include

- Version needed by GVD 1.2.0
- Update to GtkExtra 0.99.14, including a new 3D Plot widget.
- The GtkAda.Canvas now provides zooming capabilities. It also draws curve links with Bezier splines instead of circles. For efficiency, the items in the canvas are no longer double buffered. Use `Buffered_Items` if you want to keep the old behavior. `Button_Press` events are reported when an item is pressed and the event `wasn't` used to move the item around.
- New function: `Gtk.Widget.Set_Parent_Window`.
- Documentation updates

Arnaud Charlet, for the GtkAda team

[See also AUJ 22.2 (June 2001), p.73, and AUJ 21.3 (October 2000), p.163. -- dc]

## GtkAda Mailing List Update

*From: "David C. Hoos, Sr."*

*<david.c.hoos.sr@ada95.com>*

*Date: Fri, 27 Jul 2001 22:15:54 -0500*

*Subject: Re: GtkAda mailing list? (problem GtkAda on Windows)*

*Newsgroups: comp.lang.ada*

[In reply to problems with the web-pointers to the GtkAda mailing list: -- dc]

Here is a message posted yesterday to the list. It should be self-explanatory.

> My browser says that the URL for the mailing list cannot be found. Is there something wrong with your server?

We were in the process of completing the transition from `gtkada.eu.org` to `libre.act-europe.fr`. [...] Both `gtkada@gtkada.eu.org` and `gtkada@lists.act-europe.fr` should work (and be equivalent).

The archives are now hosted at `lists.act-europe.fr` (this should also be transparent).

Arno

[See also: <http://lists.act-europe.fr/mailman/listinfo/gtkada> -- dc]

## VAD 5.5 - Visual Ada Developer

*From: Leonid <dulman@attglobal.net>*

*Date: Thu, 30 Aug 2001 16:58:13 +0200*

*Subject: Announcement: Visual Ada Developer*

*VAD 5.5 Leonid Dulman*

*Newsgroups: comp.lang.ada*

VISUAL ADA DEVELOPER (VAD) version 5.5

VAD (Visual Ada Developer) is a tcl/tk oriented Ada-95 GUI builder which portable to difference platforms, such as

Windows NT/9x, Unix (Linux), Mac and OS/2.

What's new

1. New widget `html` (power `html` viewer) from `addwidgets` and `scrolled_widgets` pages
2. New widgets from `Scrolled Widgets` page: `scrolled_text`, `canvas`, `listbox`, `mclistbox` and `html`
3. From Wizard -> `Tcl Script` menu you may create window (in `canvas` or `text` ...) and use it as `scrollable frame`; build `configure script`; build `bind script`
4. Support for `XBit` package `multimedia` and `mpeg movie` (see `mpegmovie sample`); you may see all samples and work with it directly from `Image` -> `XBit` menu
5. Support for `Tkogl` package (OpenGL)
6. `XmlAda` package by Emmanuel Briot <briot@act-europe.fr>
7. Tutorial: you may run tutorial in `WEB Browser`; File `vad\tutorial\vadtutor.htm`

VAD 5.5 has four realization for `tcl/tk 8.0.x`, for `tcl/tk 8.2.3`, `tcl/tk 8.3.3` and for `tcl/tk 8.4a3` (last version).

[See also AUJ 22.2 (June 2001), pp.73-74, AUJ 22.1 (March 2001), p.13, and AUJ 21.3 (October 2000), pp.163-164, for more details. -- dc]

[...] VAD 5.5 is available in <http://www.websamba.com/dulman/vad.htm>

You may download sources `vad55scr.zip`, `vad55hlp.zip`, `adahlp.zip`, `vadtcl.zip`, `vadsmp.zip`, `adastyle.zip`, `philosophers.zip`, `vadidl.zip`, `vadtutor.zip` and binaries `vad55win.zip`, `tcl84win.zip` (WINDOWS 9x/NT) `vad55lin.zip`, `tcl84lin.zip` (OpenLinux 2.x, RedHat 6.x 7.x)

Leonid Dulman (dulman@attglobal.net)

## AWS 1.0 - Ada Web Server Component

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

*Date: 24 Aug 2001 15:51:10 +0200*

*Subject: ANNOUNCE: AWS 1.0*

*Newsgroups: fr.comp.lang.ada, comp.lang.ada*

Important note: AWS has moved on ACT Libre Software site.

A W S - Ada Web Server, 1.0 release

Authors: Dmitriy Anisimkov, Pascal Obry

August 24th, 2001

Dmitriy Anisimkov and I are very happy to announce the availability of the AWS 1.0 release. Note that the API has been changed a lot. This is part of a redesign to have a cleaner API. We plan to change slightly the API at this stage but it should be mostly stable.

Note that this is definitely a major version.

AWS stand for Ada Web Server. It is not a real Web Server like Apache. It is an HTTP component to embedded in any applications. It means that you can communicate with your application using a standard Web browser and this without the need for a Web Server. AWS is fully developed in Ada with GNAT.

Note that AWS has moved to the ACT Libre Site. Sources and documentation can be downloaded here. See pointers below.

[AWS distribution: <http://libre.act-europe.fr/--dc>]

Here are the main changes:

- The status page to use (and associated images) can be changed via the configuration file.
- Improved status page. A message is displayed if there is no session active and if there is no hotplug module loaded.
- There are now 2 initialization files: `aws.ini` (parsed first) then `<program_name>.ini`. So it is possible to have general initializations for AWS servers in the same directory and specific initializations for each AWS server.
- Now option name (in initialization files) are not case sensitive.
- Log filename is now prefixed by the name of the program (instead of "aws"). It is now possible to have many AWS program using the log facility running into the same directory.
- Forms or CGI parameters are now passed to the hotplug modules.
- Hotplug modules now support GET and POST methods.
- Client POST request fixed.
- Handle properly file name with spaces both with Netscape and IE.
- Correctly use `Upload_Path` to store uploaded files.
- A Web directory browsing services has been implemented.
- A set of icons is provided, this is used by the directory browser template.
- Remove use of AVL tree generic for session data and use a new component. The AVL tree was quite buggy. This was a serious problem, please consider upgrading to AWS 1.0 as soon as possible.
- Improve the documentation.
- Finer control of the log filename (see `AWS.Server.Start_Log` and `AWS.Log.Start`).
- Add log information in the status page.
- New log split mode policy: `Each_Run`
- Configuration options are now on a per server basis instead of per process. See

AWS.Config, AWS.Config.Set and AWS.Config.Ini.

- As always some minor bugs have been fixed but are not listed here.

[For a full announcement, including direct pointers to many modules and a list of projects using AWS, see "AWS - Ada Web Server Package" in AUJ 22.2 (June 2001), pp.75-77. -- dc]

Thanks to all who have reported bugs and have sent us patches.

Dmitriy & Pascal.

Pascal Obry, Team-Ada Member, 45, rue Gabriel Peri, 78114 Magny Les Hameaux, France  
http://perso.wanadoo.fr/pascal.obry

*From: Pascal Obry <p.obry@wanadoo.fr>  
Date: 30 Aug 2001 17:02:43 +0200  
Subject: ANNOUNCE: AWS Mailing List  
Newsgroups:*

*comp.lang.ada.fr.comp.lang.ada*

We are pleased to announce the creation of an AWS Mailing List for AWS related discussions and announces.

To subscribe go to: <http://lists.act-europe.fr/mailman/listinfo/aws>

## Ada and Internet

*From: Darren New <dnew@san.rr.com>  
Date: Fri, 24 Aug 2001 19:22:12 GMT  
Subject: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

I'm wondering how many people use Ada for Internet things. Do people write servers in Ada, or clients, or libraries? I note a lack of things like XML, MIME, base64, etc on AdaPower. If I started building such libraries, would anyone care? Would anyone use them? Just wondering...

(FWIW, I'm good at Internet stuff, and a novice at Ada, but I think I'd like Ada a lot.)

Darren New, San Diego, CA, USA (PST).

*From: Ted Dennison  
<dennison@telepath.com>*

*Date: Fri, 24 Aug 2001 19:39:52 GMT  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

Well, there's a couple of Internet related projects at <http://libre.act-europe.fr/> :

AWS, the embeddable Ada Web Server - <http://libre.act-europe.fr/aws/> and XML/Ada - <http://libre.act-europe.fr/xmlada/> .

[See "ACT-Europe - XML/Ada 0.5 Suite of Tools" in AUJ 22.2 (June 2001), p.79, for more information on the latter. -- dc]

By all means, do more. [...]

*From: Pascal Obry <p.obry@wanadoo.fr>  
Date: 24 Aug 2001 22:32:34 +0200  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

> I'm wondering how many people use Ada for Internet things.

I do. Have a look at AWS a complete Internet framework from Dmitriy Anisimkov and I. This has been used in real projects some of them stressing the server a lot with ~50 requests per second!

[And from another reply: -- dc]

As a side note, in AWS there is a base64 encoding as this is needed to transfer binary data.

*From: Kilgallen@SpamCop.net (Larry Kilgallen)  
Date: 24 Aug 2001 15:39:37 -0500  
Organization: LJK Software  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

> [...] If I started building such libraries, would anyone care? Would anyone use them? Just wondering...

Yes, people care. There is an Ada Web Server, intended not for general purpose web server purposes but rather to allow your program to serve its own pages (such as for remote control purposes).

> (FWIW, I'm good at Internet stuff, and a novice at Ada, but I think I'd like Ada a lot.)

Another data point is the degree to which people in this newsgroup fume when their web browser crashes, since they know it is quite likely to be a buffer overflow resulting from poor programming practices. Of course writing something in Ada does not automatically eliminate bugs, but it does help.

*From: "Ehud Lamm"  
<msslamm@mscc.huji.ac.il>  
Date: Sat, 25 Aug 2001 01:03:31 +0300  
Organization: The Hebrew University of Jerusalem  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

It is worth mentioning that AWS also provides for creating web clients. It is quite easy to write agents of sorts (filters, downloaders etc.). I've played with this quite a lot.

It took me some time to install the XmlAda library, but it also looks useful.

There are also methods for binding to COM objects so you can play with those.

(The GNAT Regexp tools are also useful for this problem domain).

*From: "Marc A. Criley"  
<mcqada@earthlink.net>  
Date: Fri, 24 Aug 2001 22:08:16 GMT  
Organization: Quadrus Corporation  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

> I'm wondering how many people use Ada for Internet things. [...] If I started building such libraries, would anyone care?

Please do delve into writing Internet stuff in Ada! From time to time I've whined about Ada developers announcing they're going write Yet Another Data Structures collection, while there's a whole lot of areas out there where Ada has no software presence.

Dave Botton did wonders with making Ada much easier to employ in the MS Windows environment; there's plenty of opportunity for someone to do the same with Internet technologies.

*From: Pascal Obry <p.obry@wanadoo.fr>  
Date: 25 Aug 2001 09:07:13 +0200  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

For example it would be nice to have a simple FTP client in Ada to be able to store data from time to time in a program. For example AXIS Web Camera can ftp their pictures using FTP. This is low-level Internet stuff, but quite useful... of course there are zillions of other things to do :)

[And from another message: -- dc]

It will be also very nice to have a client LDAP API. This has been on my TODO list for some time but I won't have time to work on this for a long time.

*From: "David Botton"  
<David@Botton.com>  
Date: Sun, 26 Aug 2001 11:52:05 -0400  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

It is also worth mentioning that on Win32 all the Internet tools you could want are already available from the OS through COM using GNATCOM.

Almost every day I use the MSXML object to handle all my XML processing needs. You can also control IE, use the FTP/HTTP controls that comes with the OS, etc.

It is certainly nice to have native Ada components of this nature, but in general I prefer to use the OS's capabilities when possible.

*From: "David Botton"  
<David@Botton.com>  
Date: Sun, 26 Aug 2001 21:40:22 -0400  
Subject: Re: Ada and Internet stuff  
Newsgroups: comp.lang.ada*

> Of course, but for me having native Ada components is not the point I want to have portable ones. I just don't want to be tied to a specific platform :) Also all the nice MS components are changing all the time and there is not the same ones installed on all computers... You guess, this could be a real nightmare !!

That is the beauty about COM. Code written for IE 3.0 still works today against IE 6.0. Same goes for the FTP components and just about every other component. The idea of "the moving"

APIs has not been an issue for years (at least not at the COM/DCOM level :-)

Portability is overated..... too many times I have seen sloppy products due to "portability" when it was clearly known from the start that the product would be running on a single platform. Even when a product needs to truly be cross platform, it often makes more sense to abstract out the components and APIs used and be native on each platform. A little more work, but much more professional results.

*From: tmoran@acm.org*  
*Date: Wed, 29 Aug 2001 05:52:33 GMT*  
*Subject: Re: Ada and Internet stuff*  
*Newsgroups: comp.lang.ada*

David has posted base64 encode/decode source code at [www.adapower.com/alg/base64.html](http://www.adapower.com/alg/base64.html) (That was pretty fast, eh? :)

## Ada's Web Life

*From: Terry Westley*  
*<twestley@buffalo.veridian.com>*  
*Date: Tue, 28 Aug 2001 09:04:15 -0400*  
*Subject: Re: Decline of Ada popularity*  
*To: team-ada@acm.org*

> [...] I would hope you can build 100 web sites in the same time it takes to build a single airplane. In reality, Ada has not made nearly the effort to be a super-web language as Java has, and an enormous percentage of software work is either building web stuff, or making existing stuff web enabled. CGI is fine, but does not have nearly the comfy standardized framework as Servlets, JSPs, and EJBs provide.

There's more to Ada's web life than CGI. With JGNAT, you can use Servlets, JSP, and Java Beans to create dynamic data-rich web sites. I've used the Tomcat <<http://jakarta.apache.org>> JSP server for this and have also begun using some of the other Jakarta products such as Velocity.

Advertisement: to learn more, come to SIGAda 2001 and specifically my tutorial: <<http://www.acm.org/sigada/conf/sigada2001/tutorials.html#SF2>>, "Web Application Development - Using Ada and JGNAT to develop web applications using JDBC, JSP (JavaServer Pages), Java Beans, and Java servlet technology"

Terry Westley, [twestley@acm.org](mailto:twestley@acm.org)

*From: Pascal Obry <p.obry@wanadoo.fr>*  
*Date: Tue, 28 Aug 2001 16:58:57 +0200*  
*Subject: Re: Decline of Ada popularity*  
*To: team-ada@acm.org*

And there is more than the technologies based on the JVM. There is a complete framework named AWS to develop Web applications. Used with the template engine named Templates\_Parser make it a viable choice for Web development.

AWS has already been used in some nice Web applications.

## GNADE - GNU Ada Database Environment

*From: Michael Erdmann*  
*<michael.erdmann@snaflu.de>*  
*Date: Tue, 12 Jun 2001 21:29:53 +0200*  
*Subject: \*\* Release of GNU Ada Database Development Environment \*\**  
*Newsgroups: comp.lang.ada,*  
*comp.databases.oracle.marketplace,*

GNU Ada Database Environment (GNADE) Version 1.1.4

[See also AUJ 22.2 (June 2001), p.78, and AUJ 22.1 (March 2001), pp.14-16. --dc]

This release of the GNADE project features:

- Thin Ada 95 bindings to the ODBC interface.
- Embedded ISO/92 SQL Translator generating code on top of the ODBC bindings. The translator supports dynamic and static SQL code.
- Native bindings to Postgres and MySQL.
- Installation support for MySQL, Postgres and Mimer Database products.
- Basic documentation in html, postscript and PDF format.

For more information on the project please refer to:  
<http://gnade.sourceforge.net/>

The source code distribution and a prebuild package for WindowsNT are available at:  
<http://sourceforge.net/projects/gnade>

Prebuild RPMs for Linux are available at:  
<http://www.gnuada.org/rpms313p.html>

## Ada-related Products

### DDC-I - SCORE IDE Version 2.1, Integrates PSIM PowerPC Simulator

*URL: [http://www.ddci.com/news\\_vol2num3.shtml](http://www.ddci.com/news_vol2num3.shtml)*  
*Subject: DDC-I Online News May 2001 Vol. 2 Issue 3*

SCORE Version 2.1 Integrates PSIM, A New NT And Solaris Hosted PowerPC Simulator

Phoenix, Arizona, USA, and Copenhagen, Denmark, April 6, 2001

DDC-I, a respected provider of real-time embedded system software development tools and services for safety-critical project developers since 1980, today announced the release of Version 2.1 of the proven SCORE (Safety-Critical,

Object-oriented, Real-time Embedded) suite of programming and testing tools for Ada and ANSI C.

"DDC-I has been focusing on enhancing the SCORE environment in several key areas, including the integration of PSIM, a Windows NT and Solaris hosted PowerPC simulator. PSIM allows developers to simulate a cross-development environment on a host platform, to save time and money by testing their PowerPC applications directly on their host system," explains DDC-I Engineering Manager David Mosley.

According to Mosley, SCORE's primary and ongoing mission is providing software developers an easy-to-use programming environment that guarantees maximum software portability and code reusability.

Already the first IDE (Integrated Development Environment) to offer multi-language, multi-target and multi-host capabilities based on non-proprietary open system standards, the SCORE IDE addresses the increasing need among project developers to combine reusable software components, often written in different languages, targeting different microprocessors, and created on different development platforms.

In addition to the new money-saving PSIM feature, Version 2.1 includes an improved board configuration package that allows users to customize SCORE for their specific hardware, as well as expanded support for a range of PowerPC boards, including the Excimer and the MVME 2400 series from Motorola, as well as the CVME603e from Thales Computers (formerly Cetia). By supporting several of the most popular PowerPC CPU's, PSIM also makes it a simple task to rehost to a broad range of other PowerPC boards

Other improvements to SCORE include syntax-sensitive color editing and functional debugger improvements, all within a proven IDE. The toolset includes an Ada compilation system that has full ACATS certification and a C compilation system that passes 100% of the Plum Hall test suite. All of these tools are qualified for use in the generation of certifiable code based on the FAA DO-178B and IEC 61508 standards.

The latest SCORECAST testing and validation tools for Ada and ANSI C have been integrated, with a more robust GUI created in direct response to the needs of several of DDC-I's most prominent customers like Boeing, Lockheed, and Rockwell. Each continues to migrate large safety-critical embedded system projects with software being developed by multiple vendors like Boeings innovative efforts with the

Comanche helicopter to the popular NT platform.

"SCORE is designed with features that facilitate large project management, debugging that doesn't affect the performance of embedded code, and an efficient and certifiable RTOS (Run-Time Operating System), as well as integration with the multitasking Enea OSE RTOS. Its strength is rooted in its ability to generate coherent, modular, and reusable software components, detect errors in the early stages of development, and quickly and efficiently retarget existing code," Mosley adds.

SCORE offers developers a reduction of development costs via shorter design cycles; faster time-to-market for new and redesigned products; less total programmer hours per project; and overall risk reduction through code reuse and upgradeability.

Shorter time-to-market for new applications becomes possible by allowing software development to spread across previously segregated programming groups. Risk reduction is achieved through increased component reuse and simplified target upgradeability. Because SCORE allows programmers to build using open standards Ada, ANSI C, ANDF, ASIS, POSIX, ELF/DWARF major application rewrites can be wholly avoided and dramatic reductions in overall project development costs are attainable.

The key components in SCORE are the next generation of DDC-I compilers. Based on ANDF (Architecture Neutral Distribution Format) technology, compilers for each language generate a common intermediate representation that is converted to the final object code during a later language-independent phase. Because ANDF is an XOpen standard, the SCORE system possesses a truly open architecture.

The GUI is based on Win32 and OSF/Motif, while retaining a command-line interface option, and the SCORE front-end incorporates project management tools, online help functions, tool activation, and a number of other efficient features. The GUI also offers a single universal interface for all compilers and tools, and the use of open standards means third-party products are easily integrated.

The test automation offered by SCORECAST reduces the tedious, time-related expense of module testing via consistent and repeatable processes. Offering test methodologies and automated execution for each part of the module testing process, SCORECAST also generates comprehensive test reports that facilitate improved project management and earlier defect detection.

"As SCORE continues to mature, the ANDF technology at its core will allow us to continue expanding the environments capabilities by adding features customers need quickly. We already have Ada and ANSI C ANDF front-ends (compilers), so porting existing executable code to a new target is as simple as creating a single new installer component. When the time comes to migrate legacy code, this level of flexibility can save developers more than money, it can also save them an awful lot of headaches," Mosley concludes.

## DDC-I - SCORE IDE Integrates With RTOS from Enea OSE Systems

URL: [http://www.ddci.com/news\\_vol2num3.shtml](http://www.ddci.com/news_vol2num3.shtml)

Subject: DDC-I Online News May 2001  
Vol. 2 Issue 3

Proven SCORE IDE Integrates With Leading RTOS From Enea OSE Systems  
Phoenix, Arizona, USA, and  
Copenhagen, Denmark, April 10, 2001

DDC-I, [...] today announced the integration of the proven SCORE (Safety-Critical, Object-oriented, Real-time Embedded) suite of programming and testing tools with the leading OSE RTOS (Real-Time Operating System) from Enea OSE Systems.

"The new integrated SCORE/OSE package offers software developers the ability to save both time and money. The primary purpose of the SCORE IDE (Integrated Development Environment) is to provide an easy-to-use environment that guarantees maximum code portability and reusability for safety-critical embedded system projects. Together they will generate certifiable code, based on FAA DO-178B and IEC 61508 standards, right out of the box," explains DDC-I Engineering Manager David Mosley. [...]

According to Mosley, since cost reduction is the driving force in embedded systems design, the budget benefits of SCORE/OSE don't apply exclusively to safety-critical applications, but to all real-time embedded system software development.

Enea OSE Software Manager and DER Tony Baghai explains that in recent years, many flight critical software companies that wrote their own multitasking RTOS have started turning to commercially available products like OSE. In response, Enea OSE is now developing a new FAA certifiable version as well as integrating with SCORE specifically for these safety-critical commercial and military aerospace applications.

"While integrating with SCORE, we are also upgrading our certified RTOS to include the libraries needed for Ada support, and the new release will possess full Ada capability. Millions of lines of code have been developed in Ada over the last twenty years, due to its readability, capabilities and fault tolerance, and they will need to be maintained, supported, and updated. For these reasons Ada will continue to be a significant part of the commercial and military aviation and avionics industry," he says.

A results-oriented leader in RTOS technology and development tools for constructing distributed, fault tolerant real-time embedded systems, the OSE RTOS supports robust system designs capable of recovery from hardware and software failures.

OSE's environment offers a common interface to multiple hardware platforms, and the RTOS provides several functions critical to fault tolerant safety-critical embedded system applications, the most important being memory partitioning. Allowing programs to run in parallel within individual memory partitions, programmers can create stable multitasking systems, secure in the knowledge that one application won't cause the other active programs or the larger system to crash.

Baghai adds that a number of European aviation authorities are starting to require military aircraft flying in and out of European airports to maintain the same level of software certification as commercial aircraft.

While military organizations also have stringent testing requirements, they remain different from FAA standards, and as a result many American vendors and aircraft manufacturers will face a new level of testing and verification for their military system designs, which they can address directly with the newly integrated SCORE/OSE package. Already FAA and IEC certified, SCORECAST automates module testing, eliminating the historically time-consuming and tedious process of manual testing and reporting. [...]

Available in April 2001, the first phase of integration will merge SCORE with Enea OSE's native Solaris and NT system called Soft Kernel, with a PowerPC solution immediately following. Run Time Libraries and ARINC 653 compliant partitioning will also be added. Full aerospace lifecycle services will be available, with support options ranging from certification to development and verification.

"Integrating SCORE and the OSE RTOS will provide software developers a complete Ada/C system for building fault tolerant applications, generating code



with DDC-I Ada/C's tools that easily linked to the OSE RTOS. Anyone building embedded applications, whether they're updating legacy code or migrating an entire project to a new target processor, will see bottom line results," concludes Mosley.

## DDC-I - SCORE IDE Passes Internal ACATS 2.4 Tests

URL: [http://www.ddci.com/news\\_vol2num5.shtml](http://www.ddci.com/news_vol2num5.shtml)

Subject: DDC-I Online News July 2001 Vol. 2 Issue 5

SCORE, DDC-I's Multi-language Development Environment Passes Internal ACATS 2.4 Tests

DDC-I announced today that SCORE Ada (Safety-Critical, Object-oriented, Real-time, Embedded), the Ada component of DDC-I's multi-language, software development environment, passed internal tests for ACATS (Ada Conformity Assessment Test Suite) 2.4, which is the current set of mandatory tests for obtaining a conformance certificate. SCORE is DDC-I's integrated software development environment that addresses the needs of combining reusable components, written in different programming languages, targeting a variety of microprocessors and developed on different host platforms.

One of only two Ada companies with current Ada95 ACATS certificates, DDC-I is the only company testing for Annex H (Safety & Security) which further demonstrates DDC-I's commitment to safety critical programs. Hosted on Sun/Solaris, the SCORE Ada compiler passed all tests of ACATS 2.4 for the Sun/Solaris, PowerPC and 80x86 target architectures.

One of the unique features of Ada is that Ada language processors (compilers) are validated for conformance to the Ada language standard by an outside organization (a testing laboratory). The Ada Conformity Assessment standard is an ISO standard, which defines the process that demonstrates conformity to the ISO Ada Language Standard.

The ISO Ada Language standard places requirements on five major areas of conformity assessment:

- \* The testing laboratories (Ada Conformity Assessment Laboratories - ACALs)
- \* The managing organization (Ada Conformity Assessment Authority - ACAA)
- \* The conformity assessment process
- \* The detailed procedures for conformity assessments (Ada Conformity Assessment Procedure - ACAP)

\* The test suite (Ada Conformity Assessment Test Suite - ACATS).

These requirements work together to ensure that conformity assessment is performed in a consistent and meaningful manner.

The Ada Conformity Assessment Authority (ACAA), an organization independent of the testing laboratories, is a necessary part of the assessment process. It provides the checks and balances for the testing laboratories, and ensures consistency between conformity assessments done by different laboratories.

DDC-I was first to officially pass ACATS 2.2 in 1999. DDC-I was awarded certificates A991113E2.2-001 and A991113E2.2-002 documenting this event and plans to renew the 2 year certificate according to the latest ACATS test suite.

## PegaSoft - TIA 0.7.3 Tiny IDE for Ada & TextTools 1.0.4 Console GUI

From: PegaSoft Canada - ALT Drop Box  
<adalin-1@tiamet.vaxxine.com>

Date: Sat, 30 Jun 2001 07:37:44

Subject: TIA and TextTools updated

To: "GNAT Discussion List"

<gnatlist@lyris.seas.gwu.edu>

There are new versions of TIA and TextTools located at the PegaSoft web site (<http://www.vaxxine.com/pegasoft>).

[See also AUJ 21.4 (January 2001), p.231. -- dc]

TIA 0.7.3, Tiny IDE for Ada, now has integrated CVS source control. TextTools 1.0.4, a console GUI, contains some bug fixes.

You should download both when recompiling TIA. There are also new screenshots available on the web site.

[And from another message: -- dc]

TIA is at <http://www.vaxxine.com/pegasoft/tia.html> TextTools is at <http://www.vaxxine.com/pegasoft/tt.html>

Ken O. Burtch,  
<http://www.vaxxine.com/pegasoft>,  
ken@tiamet.vaxxine.com, Pegasoft,  
R.R.#1, Jordan Station, ON, Canada L0R 1S0

## Rational - Apex Embedded 4.0.0 for Sun Solaris

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

Subject: Rational Apex Embedded Sun Solaris to Intel Architecture for Rational Exec, Intel i386/Pentium for Tornado, MIPS Family for Rational Exec, version 4.0.0 is available by FTP

To: "Apex Announcements" <apex-announcements@Rational.Com>

[Extracted from postings on Tue, 29 May 2001, Mon, 2 Jul 2001 and Thu, 19 Jul 2001. -- dc]

[In all Rational' URLs below, substitute <FTP> by ftp://ftp-stage.rational.com/public -- dc]

Product: Rational Apex Embedded for Rational Exec Version: 4.0.0 Platform: Sun Solaris to Intel Architecture URL: <FTP>/apex\_cross/releases/sol/i386/rt/4.0.0

Product: Rational Apex Embedded for Tornado Version: 4.0.0 Platform: Sun Solaris to Intel i386/Pentium URL: <FTP>/apex\_cross/releases/sol/i386/vw/4.0.0

Product: Rational Apex Embedded for Rational Exec Version: 4.0.0 Platform: Sun Solaris to MIPS Family URL: <FTP>/apex\_cross/releases/sol/mips/rt/4.0.0

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 [the product] download and installation instructions. There are 3 ways to do the download. You can use the classic method of the UNIX ftp command, use a web browser, or use Rational's rinstall program to do the FTP download in a user-friendly way. <FTP>/standard.msgs/install\_instructions.html

Release Notes and Install Guide are located here:

[In these URLs, substitute <DOC> by <FTP>/apex\_cross/documents/unix -- dc]

Rational Apex Exec 4.0.0 Release Note for i386: <DOC>/Rexec/relnote.4.0.dir/rexec\_relnote\_386/rexec\_release\_noteTOC.html  
<DOC>/Rexec/relnote.rexec.386.4.0.ps.Z

Rational Apex Embedded 4.0.0 Release Note for i386: <DOC>/VxWorks/relnote.4.0.dir/vxworks\_relnote\_386/vxworks\_release\_noteTOC.html  
<DOC>/VxWorks/relnote.vxworks.386.4.0.ps.Z

Rational Apex Exec 4.0.0 Release Note for MIPS: <DOC>/Rexec/relnote.4.0.dir/rexec\_relnote\_mips/rexec\_release\_noteTOC.html  
<DOC>/Rexec/relnote.rexec.mips.4.0.ps.Z

Eddie Glenn, Product Manager, Rational Software

## Rational - Apex 4.0.0c for SGI IRIX & HP HP-UX

From: "Greg Bek" <gab@Rational.Com>



*Subject: Rational Apex 4.0.0c for SGI IRIX & HP HP-UX is available by FTP*  
*To: "Apex Announcements" <apex-announcements@Rational.Com>*

[Extracted from postings on Wed, 6 Jun 2001 and Tue, 17 Jul 2001. -- dc]

Product: Rational Apex Version: 4.0.0c  
 Platform: SGI IRIX URL:  
 <FTP>/apex/releases/sg32/apex.4.0.0c

Product: Rational Apex Version: 4.0.0c  
 Platform: HP HP-UX URL:  
 <FTP>/apex/releases/hp/apex.4.0.0c

This is the Generally Available (GA) release [for the SGI platform]. This release [for the HP platform] 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 download and installation instructions. There are 3 ways to do the download. You can use the classic method of the UNIX ftp command, use a web browser, or use Rational's rinstall program to do the FTP download in a user-friendly way.  
 <FTP>/standard.msgs/install\_instructions.html

[The SGI] version of Apex requires new license keys with a version of 4.000. Please contact Rational Technical Support for assistance.

The Release Notes and Installation Guide for Apex 4.0.0c can be found by following the links listed below. [...]

[In these URLs, substitute <DOC> by <FTP>/apex/documents/unix -- dc]

Rational Apex 4.0.0c Release Note:  
 <DOC>/release\_note.4.0.0c.dir/release\_noteTOC.html  
 <DOC>/release\_note.4.0.0c.ps.Z

Rational Apex 4.0.0c Install Guide:  
 <DOC>/install\_guide.4.0.0c.dir/igtOC.html  
 <DOC>/install\_guide.4.0.0c.ps.Z

Greg Bek, Product Manager, Rational Software, Cupertino CA 95014, Phone: +1 408 863 4394, Fax: +1 408 863 4180

## Rational - Apex 4.0.0b Patches for Solaris 64 bit support

*From: "Greg Bek" <gab@Rational.Com>*  
*Date: Thu, 7 Jun 2001 11:07:27 -0700*  
*Subject: Patches for Apex 4.0.0b - Solaris 64 bit support*  
*To: "Apex Announcements" <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

200105161-1 - 64 bit Ada Compiler and Runtimes  
 200105162-1 - 64 bit Ada predefines  
 200105163-1 - 64 bit C++ Compiler and Models

[...]

The patches are located at:  
 <FTP>/apex/releases/sol/patches/Ada/200105161-1 (41MB)  
 <FTP>/apex/releases/sol/patches/Ada/200105162-1 (182MB)  
 <FTP>/apex/releases/sol/patches/C++/200105163-1 (24MB)

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

## Rational - AXI (Ada/X Interface) 4.1.9 & Ada Analyzer 4.0.1, for SGI IRIX & HP HP-UX

*From: "Greg Bek" <gab@Rational.Com>*  
*Subject: AXI (Ada/X Interface) 4.1.9 & Ada Analyzer 4.0.1 for SGI IRIX & HP HP-UX is available by FTP*  
*To: "Apex Announcements" <apex-announcements@Rational.Com>*

[Extracted from postings on Wed, 6 Jun 2001 and Tue, 17 Jul 2001 -- dc]

Product: AXI (Ada/X Interface) Version: 4.1.9  
 Platform: SGI IRIX URL:  
 <FTP>/axi/releases/sg32/axi.4.1.9

Product: AXI (Ada/X Interface) Version: 4.1.9  
 Platform: HP HP-UX URL:  
 <FTP>/axi/releases/hp/axi.4.1.9

Product: Ada Analyzer Version: 4.0.1  
 Platform: SGI IRIX URL:  
 <FTP>/ada\_analyzer/releases/sg32/ada\_analyzer.4.0.1

Product: Ada Analyzer Version: 4.0.1  
 Platform: HP HP-UX URL:  
 <FTP>/ada\_analyzer/releases/hp/ada\_analyzer.4.0.1

This is the Generally Available (GA) release [for the SGI platform]. This

release [for the HP platform] 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 [the product] download and installation instructions. There are 3 ways to do the download. You can use the classic method of the UNIX ftp command, use a web browser, or use Rational's rinstall program to do the FTP download in a user-friendly way.  
 <FTP>/standard.msgs/install\_instructions.html

Release Notes and Install Guide are located here:

[In these URLs, substitute <DOC> by <FTP>/apex/documents/unix -- dc]

AXI 4.1.9/IRIX N32 Release Note:  
 <DOC>/axi\_relnotes.4.1.9.sg32.dir/relnotesTOC.html  
 <DOC>/axi\_relnotes.4.1.9.sg32.ps.Z

Ada Analyzer 4.0.1 Release Note:  
 <DOC>/release\_note.4.0.1.dir/AA\_Release\_Note.4.0.html  
 <DOC>/release\_note.4.0.1.ps.Z

## Rational - Apex 4.0 Certificates of Conformity

*From: "Greg Bek" <gab@Rational.Com>*  
*Date: Tue, 19 Jun 2001 14:22:09 -0700*  
*Subject: Apex 4.0 Certificates of Conformity*  
*To: "Apex Announcements" <apex-announcements@Rational.Com>*

In May, Rational conducted on-site witness testing of 8 Apex 4.0 variants against the new ACATS 2.4 test suite.

The Certificates of Conformity (formerly called Validation Certificates) can be viewed online at our main web site:  
 http://www.rational.com/products/apex/certificates.jsp

Please note that not all Apex variants are listed with current certificates. As we release more Apex 4.0 products we will conduct further witness testing and post certificates to the web site.

Note also that not all products listed with certificates are available for shipping from Rational. On some platforms the compilation system was been certified, but we are still in the process of building and testing the entire release which includes layered products such as TestMate, Ada Analyzer etc.

New certificates are: #A010516E2.4-001 SPARC Solaris #A010516E2.4-002 Silicon Graphics IRIX #A010516E2.4-003 Digital Alpha AXP/UNIX #A010516E2.4-004 SPARC Solaris to PowerPC with Rational Exec

#A010516E2.4-006 HP 9000 700/800 HP\_UX (Not yet shipping)  
 #A010516E2.4-005 SPARC Solaris to PowerPC with Tornado #A010516E2.4-007 SPARC Solaris to PowerPC with LynxOS #A010516E2.4-008 SPARC Solaris to Intel x86 with Rational Exec

## Ada and Linux

### New Linux packages on www.gnuada.org

*From: Jürgen Pfeifer*  
*a<juergen.pfeifer@gmx.net>*  
*Date: Sun, 24 Jun 2001 22:56:18 +0200*  
*Subject: New versions uploaded*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

I just completed the upload of these updated packages for GNU/Linux x86 based systems using the RPM package format:

<http://www.gnuada.org/rpms313p.html#GTKADA> and <http://www.gnuada.org/rpms313p.html#GVD>

*From: Jürgen Pfeifer*  
*<juergen.pfeifer@gmx.net>*  
*Date: Wed, 27 Jun 2001 17:04:55 +0200*  
*Subject: RPM for adabroker uploaded*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

I've uploaded a new RPM containing the Pre4 Beta of adabroker version 1.0.

See <http://www.gnuada.org/rpms313p.html#ADABROKER>

*From: PegaSoft Canada - ALT Drop Box*  
*<adalin-1@tiemet.vaxxine.com>*  
*Date: Sun, 8 Jul 2001 13:39:47 -0400 (EDT)*  
*Subject: TIA RPMs updated*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

Many thanks to Juergen for updating the TIA RPMs! His help is much appreciated. The RPMs are available on the ALT download page.

Ken O. Burch

## Ada and Microsoft

### GNAT 3.12p/3.13p for DOS

*From: "Victor Shkammerda"*  
*<vvs@auto.bnm.org>*  
*Date: Thu, 07 Jun 2001 10:19:29 +0300*  
*Subject: ANNOUNCE: GNAT/DOS 3.13p source patches available at SourceForge*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

I've just uploaded the patches and build scripts which I've used to make a GNAT/DOS 3.13p binary release. It should appear at <ftp://download.sourceforge.net/pub/>

[sourceforge/gnuada/gnat313s.zip](http://sourceforge.gnuada/gnat313s.zip) shortly. No docs yet, in the mean time you might want to look at the scripts (in particular at [install.bat](#)). You will need sources of GCC 2.8.1 and GNAT 3.13p, which you can download from <ftp://ftp.gnu.org/pub/gnu> and <ftp://cs.nyu.edu/pub/gnat>. Also, you will need DJGPP 2.03, which you can get from <ftp://ftp.simtel.net/pub/simtelnet/gnu/djgpp>.

*From: "Gautier de Montmollin"*  
*<gdemont@hotmail.com>*  
*Date: Tue, 10 Jul 2001 10:03:13 -0000*  
*Subject: Re: GNAT for DOS?*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

As communicated privately, the 3.12 files for DOS are now in their new place. Enjoy ;-)

[See <http://www.mysunrise.ch/users/gdm/gnatdos.htm> -- dc]

### SETI@Home Service 2.1

*From: Ted Dennison*  
*<dennison@telepath.com>*  
*Date: Fri, 29 Jun 2001 13:49:47 GMT*  
*Subject: ANN: SETI@Home Service version 2.1*  
*Newsgroups: comp.lang.ada*

Version 2.1 of the SETI@Home Service is now available for download at [http://www.telepath.com/dennison/Ted/SETI/SETI\\_Service.html](http://www.telepath.com/dennison/Ted/SETI/SETI_Service.html).

The major innovation in this version is the ability to designate processes for which the SETI@Home clients will be suspended or killed. The clients will automatically be restarted when the designated process(es) complete their execution.

To accomplish this, a new package has been added to the thick Windows bindings: Performance\_Information. This package encapsulates the ability to query the NT/Win2k registry for any kind of real-time performance information. Pretty much any information available via the Windows Task Manager or "perfmon" must be read in this way (and its a \*bear\* to do, let me tell you...) I only needed the ability to get the names of the running processes, so this facility isn't quite complete (and probably isn't yet much good for anything else). But for the next SETI@Home Service version I plan to add detection of CPU load, which will require me to make this package much more complete.

This version also upgrades the AWS support to AWS 0.10 (that's "zero point ten" :-)), and includes one new routine in the Registry package. I suspect that the SETI@Home Service may be the most widely used AWS program in existence.

Statistical digression: I'm not metering downloads, but the entire SETI@Home project has over 3 million users, 500,000

of which are considered "active". Over 300 million results have been submitted, about 230 million of which were submitted by Wintel boxes. 50 million of those were submitted by the command-line client (which is the client "serious" users prefer, due to its speed). Presumably most of these users are not programmers, although probably a fair amount of the "serious" users are. (Indeed most of the emails I get about the project are from non-programmers :-)).

[...] For more information, see the website at [http://www.telepath.com/dennison/Ted/SETI/SETI\\_Service.html](http://www.telepath.com/dennison/Ted/SETI/SETI_Service.html)

[and also "SETI Service 1.0 - Creating a NT service using Ada" in AUJ 21.4 (January 2001), p.234. -- dc]

### On SETI@Home Service's Use of Ada on the Windows Platform

*From: Ted Dennison*  
*<dennison@ssd.fsi.com>*  
*Date: Wed, 15 Aug 2001 08:36:20 -0500*  
*Subject: RE: Has anyone experimented with DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

[In a thread about working platform independent: -- dc]

There are often \*other\* reasons for being platform specific. For instance, the SETI@Home Service is platform specific to NT and Win2k, because that's the whole point of the project. If you wanted to have something that is always running on Unix even when no-one is logged in, you'd have to use a completely different mechanism. Given that its already tied to a single platform by its requirements, it makes perfect sense to be calling NT-specific system services (eg: the registry calls, registry-based real-time task information).

Likewise, there are plenty of reasons to use Ada other than its improved portability. I don't think anyone here would seriously make the argument that I should have used VisualC++ to code the SETI@Home Service, because it was going to be non-portable anyway. For instance, I think the safety of the language is a big selling point for serious SETI users. Any bug is liable to cost them precious processing time (and thus perhaps a ranking).

*From: Ted Dennison*  
*<dennison@ssd.fsi.com>*  
*Date: Wed, 15 Aug 2001 10:23:38 -0500*  
*Subject: RE: Has anyone experimented with DirectX (Direct Draw)*  
*To: "GNAT Discussion List"*  
*<gnatlist@lyris.seas.gwu.edu>*

>Does the SETI project use Ada?

I do not know what language the SETI@Home project uses for their client

software. If its Ada, I'd be (pleasantly) surprised. However, the SETI@Home Service is a project that I run that allows Windows NT and 2k users to run a SETI@Home client in an NT service so that it can run whenever the machine is booted (whether or not anyone is logged in). That project is indeed done entirely in Ada. For more info on it, see the website at [http://www.telepath.com/dennison/Ted/SETI/SETI\\_Service.html](http://www.telepath.com/dennison/Ted/SETI/SETI_Service.html).

## Claw.Sockets Package

*From: t Moran <t Moran@acm.org>  
Date: Mon, 27 Aug 2001 18:46:24 GMT  
Subject: Re: My Gnat is rusty - link error  
Newsgroups: comp.lang.ada*

[A compiler independent alternative for the GNAT.Sockets package. -- dc]

I'd modestly offer the Claw.Sockets interface (see [www.rsoftware.com](http://www.rsoftware.com)). It works on multiple compilers targeted to Windows, and the actual Windows dependencies are inside the implementation, mostly concerning asynchronous stuff that uses the Windows message loop. An example very simple web server (ca 400 lines) using it is on [www.adapower.com](http://www.adapower.com)

## References to Publications

### DDC-I Online News

[See elsewhere in this AUJ issue for selected items. -- dc]

*From: JC <jcdk@ddci.com>  
Date: Thu, 24 May 2001 14:40:37  
Organization: DDC-I  
Subject: DDC-I Online News  
To: N8 DK Online Newsletter  
<jcdk@ddci.com>*

DDC-I Online News May 2001, Vol. 2, Num. 3 A monthly news update dedicated to DDC-I customers & registered subscribers.

This Month:

\* DDC-I announces the integration of the proven SCORE (Safety-Critical, Object-oriented, Real-time Embedded) suite of programming and testing tools with the leading OSE RTOS (Real-Time Operating System) from Enea OSE Systems. The new integrated SCORE/OSE package offers software developers the ability to generate certifiable code, based on FAA DO-178B and IEC 61508 standards, right out of the box.

\* SCORE Version 2.1 Integrates PSIM, a new NT and Solaris hosted PowerPC simulator. PSIM allows developers to simulate a cross-development environment on a host platform, to save time and money by testing their

PowerPC applications directly on their host system.

[...] For the complete newsletter, go to [http://www.ddci.com/news\\_vol2num3.shtml](http://www.ddci.com/news_vol2num3.shtml) [...]

*From: JC <jcdk@ddci.com>  
Date: Wed, 27 Jun 2001 16:44:27  
Organization: DDC-I  
Subject: DDC-I Online News  
To: O8 June2001 DK Online News  
<jcdk@ddci.com>*

DDC-I Online News June 2001, Vol. 2, Num. 4 A monthly news update dedicated to DDC-I customers & registered subscribers.

This Month:

\* Embedded, Real-time, Safety-critical Control is Nothing New to the Ada Language - As the world moves toward more and more automated processes the use of industrial controllers is increasing, as is the scope of their functional specifications. One requirement of increasing importance is the need for these controllers to meet safety-critical standards due to their use in processes where a failure could result in serious harm to individuals. Read more on this topic, as published in the May issue of Control Solutions magazine. If you have safety-critical software requirements, we guarantee you will find this interesting.

\* TADS-i960 Keeping Apache Pointed In The Right Direction - Hamilton-Sundstrand's Dynamic Controls division relies on DDC-I's TADS toolset to upgrade and support the AH-64D Apache helicopter flight control module.

[...] For the complete newsletter, log on to [http://www.ddci.com/news\\_vol2num4.shtml](http://www.ddci.com/news_vol2num4.shtml) [...]

*From: JC <jcdk@ddci.com>  
Date: Tue, 31 Jul 2001 14:37:43  
Organization: DDC-I  
Subject: DDC-I Online News  
To: R8 July 2001 DK Online News  
<jcdk@ddci.com>*

DDC-I Online News July 2001, Vol. 2, Num. 5 A monthly news update dedicated to DDC-I customers & registered subscribers.

This Month:

\* Mark 41 Vertical Launch System Depends On DACS. Lockheed Martin leans on DDC-I's Proven DACS-80x86 development tools to minimize maintenance costs for the safety-critical core of the world-class Mark 41 Vertical Launch System.

\* DDC-I's Multi-language Development Environment Passes Internal ACATS 2.4 Tests. An informative look into this ISO standard, which defines the process that demonstrates conformity to the ISO Ada Language Standard.

[...] For the complete newsletter, go to [http://www.ddci.com/news\\_vol2num5.shtml](http://www.ddci.com/news_vol2num5.shtml) [...]

## Ada articles in May 2001 issue of "COTS Journal"

*From: Clyde Roby <roby@ida.org>  
Date: Fri, 8 Jun 2001 11:55:20 -0400  
Subject: Ada articles in May 2001 issue of "COTS Journal"  
To: team-ada@acm.org*

The May 2001 issue of COTS Journal has "Ada Finds New Life in Commercial Application and Legacy Upgrades" displayed on the cover.

There are two Ada-related articles: "Beyond C++ and Java: Extending the Reach of Ada Through CORBA" by Vic Giddings of Objective Interface Systems (OIS) [6 pages], and "Ada Stands Ready for the 21st Century" by Joyce Tokar of DDC-I [5 pages].

[See also "Ada 95 accelerates time to market - Article in COTS magazine" in AUJ 21.4 (January 2001), p.235. -- dc]

*From: "Thomas A. Panfil"  
<t.panfil@gte.net>  
Date: Sat, 9 Jun 2001 15:10:20 -0400  
Subject: Re: Ada articles in May 2001 issue of "COTS Journal"  
To: team-ada@acm.org*

Their web site is posting issues up to March 2001 now. In a couple months, check out: <http://www.rtcgroup.com/cotsjournal/index.shtml>

Tom Panfil, Baltimore SIGAda

[These articles are available now at <http://www.rtcgroup.com/cotsjournal/may2001/onthesofterside.pdf> -- dc]

## "C++ Falls Short of an 'A'" - from "eWeek" Magazine

*From: rdriehle@nps.navy.mil (Riehle, Richard)  
Date: Wed, 13 Jun 2001 18:06  
Subject: Peter Coffee Strikes Again  
Newsgroups: comp.lang.ada*

There is an entertaining column in eWeek (June 11, 2001, page 47) with the title, "C++ Falls Short of an 'A,'" by Peter Coffee. He has some kind words for Ada as well as for Java and Modula-2. [...]

<http://www.eweek.com>

*From: Ted Dennison  
<dennison@telepath.com>  
Date: Wed, 13 Jun 2001 18:40:09 GMT  
Subject: Re: Peter Coffee Strikes Again  
Newsgroups: comp.lang.ada*

[...] I loved this comment:

"Is it an advantage that more C++ programmers can always be found when

needed? Or is it a serious problem that you always seem to need one more?"

URL: <http://www.zdnet.com/eweek/stories/general/0,11011,2769111,00.html>

eWEEK Commentary

C++ falls short of an 'A'

By Peter Coffee June 11, 2001

[A few extracts follow, you should read the whole (copyrighted) article online: -- dc]

College computer science programs will have the task, during the next few years, of digesting a cohort of incoming students who learned C++ as their first programming language. [...] The college-level advanced placement computer science course switched from Pascal to C++ in 1999-and has finally seen the error of its ways. [...]

I'll repeat what I said about C++ in 1994: "The programming language's 'feature'-being a super set of C-is a fundamental bug." The College Board report agrees, citing the inability of C++ to prevent or handle errors in using arrays. Industrial advisory boards also agree, for example, recommending Ada or Modula-2 ("having fewer insecurities and better type checking") for writing the software underlying automotive systems.

The College Board's short-lived adoption of C++ exemplifies a far more general IT management error: the assumption that what's popular must be good. [...]

---

## Ada Inside

### World-wide - Cable TV Applications

From: "Marin David Condic"

<[marin.condic@pacemicro.com](mailto:marin.condic@pacemicro.com)>

Date: Wed, 13 Jun 2001 09:27:56 -0400

Subject: A new market in which Ada can play?

Newsgroups: *comp.lang.ada*

For those interested in promoting Ada and finding new markets in which Ada can be a player, here's an emerging opportunity:

<http://www.linuxdevices.com/news/NS9331678324.html>

[The article at that URL is entitled: "Two dozen companies to bring Linux to TV set-tops". -- dc]

Since all if this is a brand new horizon (Cable TV applications) there is no built-in body of investment in other development technologies. Hence the only thing to overcome is any anti-Ada bias that may exist in the minds of techies. Considering the impact of any software errors on interruption of service, etc., Ada could have some really strong selling points in terms of reliability. Since

it is all Linux and GNU, etc, it should not be a big shift for Ada to be a player - it just needs some attention paid to it by compiler vendors, app developers, etc.

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

Date: 13 Jun 2001 17:58:02 +0200

Subject: Re: A new market in which Ada can play?

Summary: *Ada is already playing there!*  
Newsgroups: *comp.lang.ada*

> Since all if this is a brand new horizon (Cable TV applications) [...]

It might be interesting to check out the following URL:  
<http://www.cs.kuleuven.ac.be/conference/AE2001/program.html#speak> and read the second abstract.

As you can see there, using Ada in Cable TV applications is not completely new: the abstract at the URL above is about the keynote presentation that Pascal Heraud from CANAL+ Technologies gave at the recent Ada-Europe'2001 conference in Leuven 4 weeks ago, entitled:

"Using Ada in Interactive Digital Television Systems"

CANAL+ has been using Ada in their server-side components since quite some time. People from that company are reading *comp.lang.ada* (hi Thierry!) and might be tempted to comment on the news Marin refers to?

[See also "Word-wide / MediaOne & Canal+ - TV Head-ends Systems" in AUJ 21.1 (April 2000), pp.36-37. -- dc]

It would of course be great if Ada would be used by other companies in this market and it could be very valuable in the TV set-top boxes too.

The full paper of the above-mentioned keynote presentation is printed in the Ada-Europe'2001 conference proceedings (together with the other 4 invited papers and the 27 technical papers). The reference is:

Dirk Craeynest and Alfred Strohmeier (Eds.): *Reliable Software Technologies - Ada-Europe 2001: Proceedings*; 6th Ada-Europe International Conference on Reliable Software Technologies; Leuven, Belgium, May 2001; Lecture Notes in Computer Science, vol. 2043; Springer-Verlag, 2001; ISBN 3-540-42123-8

These proceedings are available from every good bookshop, or can be ordered directly from Springer:

[http://www.springer.de/cgi-bin/search\\_book.pl?isbn=3-540-42123-8](http://www.springer.de/cgi-bin/search_book.pl?isbn=3-540-42123-8)

They are also available on-line on Springer's LINK website:  
<http://link.springer.de/link/service/series/0558/tocs/t2043.htm>

The table of contents and all abstracts are freely readable on-line, but access to the copyrighted full papers is paying.

Dirk Craeynest, Ada-Europe'2001 Program Co-Chair

PS: Note that, except for being co-editor of these proceedings, I have no bindings with Springer, and have no financial benefit in the sale of this book.

Dirk Craeynest, Offis - Aubay Group, Weivedlaan 41/32, B-1930 Zaventem, Belgium, Email

[Dirk.Craeynest@cs.kuleuven.ac.be](mailto:Dirk.Craeynest@cs.kuleuven.ac.be) (AE'2001), [Dirk.Craeynest@offis.be](mailto:Dirk.Craeynest@offis.be) (work), Phone +32(2)725.40.25, Fax +32(2)725.40.12

### Embedded, Real-time, Safety-critical Control is Nothing New to the Ada Language

[From: "DDC-I Online News, June 2001, Vol. 2 Issue 4. -- dc]

URL: [http://www.ddci.com/news\\_vol2num4.shtml](http://www.ddci.com/news_vol2num4.shtml)

Embedded, Real-time, Safety-critical Control is Nothing New to the Ada Language

With respect to software, the world of industrial controllers is unique due to the need to produce programs that execute on embedded processors, often in hard real-time, with limitations imposed by the execution environment. For example, controllers used to run a steel making process must manage the task in a timely fashion while operating in intense heat. As the world moves toward more and more automated processes the use of industrial controllers is increasing, as is the scope of their functional specifications. One requirement of increasing importance is the need for these controllers to meet safety-critical standards due to their use in processes where a failure could result in serious harm to individuals.

There are a number of steps involved in meeting the growing requirements for industrial controllers while at the same time developing a process that optimizes the development cycle. Of primary concern is the selection of a programming language that can support the expanding needs of the industrial controller community.

Some important features of the programming language are standardization, multitasking, object-oriented programming, code reuse, support for safety-critical applications, and demonstrated success in the development of industrial control software. Many of the existing high-level languages available today satisfy some of these characteristics. The Ada programming language is the only

commercially available programming language that meets all of these criteria.

There may be some question as to the value of using a high-level programming language for the development of software for industrial controllers. Until recently, most of this software was developed in assembly language. However, using a high-level programming language promotes reuse of software components. In addition, high-level languages provide more expressive means to describe an algorithm. This aids in the readability and long-term maintenance of the software. Finally, in those instances where it is most effective to use assembly language to implement an algorithm or to access the hardware, Ada supplies features that allow for the smooth interaction of assembly language modules with those developed in Ada.

Using a high-level programming language that supports all of the characteristics of industrial controller software will reduce the cost of development both in time and money. Expanding on each of these criterion individually will demonstrate why Ada is the best choice.

Ada was designed for real-time, embedded system applications such as mission-critical systems in the military. Early in Ada's lifetime, it was apparent that this language was well suited to a number of safety-critical embedded applications beyond the military. Its use extended into commercial flight control and a variety of process control applications. With the introduction of full support for object-oriented programming in the revised Ada standard in 1995, the use of the language for embedded applications such as industrial control systems has expanded.

The Ada programming language is an ISO standard that is maintained through a collection of working groups that represent the global Ada community. The standard provides for a consistent definition of the programming language that is in use worldwide. In addition, an independent test suite has been developed and maintained to assure the conformance of an Ada compilation system that also meets the standard. In this manner, the tools and the application code adhere to a common standard that is used internationally. The independent test suite also lends itself to the compilation system validation and verification process common to many safety-critical development environments.

Ada supports a full tasking model within the core definition of the language. This model is designed to interoperate smoothly with the object-oriented features of the language such as dispatching and inheritance. The tasking model provides full task synchronization, which supports the coordinated exchange

of data between threads as parameters. This model also offers data-oriented synchronization to facilitate asynchronous and protected exchange of data between threads. In addition, the language standard includes the definition of specific profiles to support the use of the tasking model in real-time applications. The standard goes on to define a domain-specific profile for high-integrity systems.

In its original definition, Ada provided data structures and semantics for object-oriented design. As part of the five-year upgrade to the language standard, full support for object-oriented programming was added. The model supports the development of classes of objects with single-inheritance and polymorphism. These features are tightly coupled with all of the features in the language, supporting the construction of real-time, object-oriented applications.

Reuse was a primary objective in the definition of the Ada programming language. There are a variety of features for the generation of software components. Two key items that are useful in the development of reusable software are packages and generics. For example, data abstraction is supplied through the package structure. This construct may be used to supply a client with a clear definition of the interface to the module without exposing the actual implementation of the module. The hierarchical library extends the model by providing a number of different module bodies that may be used for a given interface.

Generics are templates similar to macros in assembly language. A generic module is defined with parameters for data and operations. A module for a specific set of data and operations is created by instantiating the generic with the appropriate attributes. Generics may be used in industrial control programs to supply a common algorithm to a number of different process paths, each of which may be using different data streams. All of these language features were designed to coexist. Hence, the language supports multitasking, object-oriented programming including generics. So a user may define a generic that may be used as a template to define class hierarchies within a threaded process architecture.

In some industrial control systems, error conditions in the environment are programmed as exceptional events in the application, and appropriate action is to be taken when such an event occurs. Ada incorporates exception handling capabilities in the language that operate with the multithreading and object-oriented features of the language. Hence, exceptional conditions may be handled locally within a single thread of

execution or propagated to a higher level to handle the graceful termination of all threads of execution.

Early error detection is a key component in reducing the costs of a program. Errors that are detected at the start of a project are much less expensive to correct than those found during system integration and testing. Ada requires early detection of type conflicts and inappropriate use of data. Hence, the compilation system checks for out-of-range values and type mismatches as part of the preliminary processing of the source code. Many errors are detected prior to the generation of code. Once code has been generated, there are automatic range checks that will generate an error if an inappropriate value is assigned to a variable. This feature is particularly useful when interfacing with modules developed in other languages such as assembly.

#### Conclusion

Ada offers the industrial control community a path to the future that is derived from proven embedded systems applications. It is designed to be cost effective in generating the desired results for real-time, industrial control applications in an optimal fashion.

This article also published in the May 2001 issue of Control Solutions Magazine.

## USA / Hamilton-Sundstrand - AH-64D Apache Helicopter Flight Control Module

[From: "DDC-I Online News, June 2001, Vol. 2 Issue 4. -- dc]  
URL: [http://www.ddci.com/news\\_vol2num4.shtml](http://www.ddci.com/news_vol2num4.shtml)

### TADS-i960 Keeping Apache Pointed In The Right Direction

Hamilton-Sundstrand's Dynamic Controls division relies on DDC-I's TADS toolset to upgrade and support the AH-64D Apache helicopter flight control module

In September 1999, the AH-64D Apache Longbow, the world's only fourth-generation attack helicopter, performed a series of loops, rolls, hammerhead stalls, and split-S maneuvers at weights never before flown in public demonstrations. The dramatic series of aerobic maneuvers showcased its flight performance, agility, and an unmatched ability to provide combat pilots a decisive edge in the sky.

Reporting that the helicopter performed even better than its predecessor, the combat-proven AH-64A Apache, Boeing pilots also confirmed the rock-solid performance of the Apache Longbow's advanced avionics suite, including the safety-critical embedded systems within the primary flight controls, designed and

built by DCHS, the Dynamic Controls division of Hamilton-Sundstrand.

A recognized industry leader in design and development of electronic and electro-mechanical controls and systems for the military and commercial aerospace markets, Dynamic Controls was acquired in 1996 by then Hamilton Standard, which was merged with Sundstrand by parent United Technologies in 1999.

Also presented NASA's highest recognition for vendor quality in 1996, the George M. Low award, for its contributions to the space program, Hamilton has designed and built an impressive collection of high-profile systems. From the life support and environmental controls onboard Apollo 11 to multiple systems flying on the maiden voyage of the Space Shuttle Columbia, as well as over 100 components currently orbiting in the International Space Station, DCHS programmers genuinely understand the meaning of safety-critical.

"The safety-critical aspects of real-time embedded systems developed by DCHS are addressed right from the beginning on every project, in the system requirements phase," says Kate Goldstone, the Senior Engineer responsible for the systems and software groups within the flight control enterprise at DCHS.

Goldstone explains that the Ada programming language and the Tartan Ada Development System (TADS) are used to create the embedded code targeting the i960 processor in a recently upgraded flight control module. The Apaches previous module, designed a decade ago, contains the earlier ceramic version of the Intel chip, which is no longer manufactured and has since been replaced by a plastic version.

"The Apache program is obviously a mature one, and what we were actually doing was performing an upgrade on the system due to obsolescence issues with the old processor," she says.

According to Goldstone, Hamilton purchased their initial TADS toolset from Tartan and stayed with the original delivery for eight years, never making contact with Texas Instruments during its tenure with TADS. Not until 1999, when the Apache flight control module underwent a complete redesign, did they purchase the latest version of TADS-i960 from DDC-I.

"We ended up with a couple cases where changes [over the eight year interval] caused some features we had utilized in the past not to work very well. Being able to specify where code would go based on memory map I/O, that was our biggest issue. We've been able to get around those things, but I'll admit it was a little

painful, and a little time consuming, to get through the conversion," she adds.

However, Goldstone takes a pragmatic approach to what the flight control team at DCHS went through to bring the TADS toolset up to speed.

"I think its a normal thing when you change compilers and vendors to find some new quirks, and I think when DDC-I acquired the system from Texas Instruments, there wasn't a lot of knowledge transfer from TI. What I found was that it was a functional product, but DDC-I was playing catch-up to discover what was wrong with the system and set it straight," she comments.

An expert in Ada development for safety-critical real-time embedded systems, DDC-I acquired Tartan from TI in early 1998, to round out their arsenal of Ada development systems. Confirming Goldstone's sense of the product, TADS had been largely unsupported under TI ownership, and DDC-I's first goal for the mature TADS products targeting the venerable Intel i960, Motorola 68XXX, and MIL-STD-1750A processors was to beef up engineering services and support.

Headquartered in Lyngby, Denmark and Phoenix, Arizona, DDC-I is a long-standing developer of Ada systems and tools, from compilers, run time systems, and debuggers to fully integrated development environments. DDC-I also offers consulting and support services, maintaining a distinguished list of civil, commercial, and defense clients, including Boeing, Sikorsky, and Lockheed-Martin.

The flight controls enterprise at DCHS also serves a who's who in the civil, commercial, and defense aerospace industries, and Goldstone's role is pivotal to the timely completion of each project. Responsible for scheduling, staffing, and budgets for every program within the group, including the flight controls and vehicle management for the Lockheed-Martin Joint Strike Fighter, her position affords her a birds eye view of the safety-critical embedded systems industry and the challenges confronting contractors today.

"If you've developed per mil-standard in the past you've got all of your processes and certifications in place and you stick with the game plan," she says.

She explains that developers working on embedded systems for military projects have no choice but to have proven processes in place, while in recent years there's been a trend on the commercial side in the US which started much earlier in Europe toward developing better process framework, due to a few high-profile failures as a result of ad hoc development. As an experienced military developer, all DCHS safety-critical design processes flow in accordance with

a thoroughly tested design development manual.

Support is also a completely different beast in the defense sector according to Goldstone, which is why DCHS chose to contact DDC-I about upgrading their TADS environment. Under contract, they are required to support the upgraded Apache flight control systems until 2010.

The Ultra 60 running Solaris that hosts the TADS toolset is a modern platform, but the computing platforms at DCHS run the gamut, due to their obligation to support programs until the contract expires. While they also use NT and other contemporary systems, they still maintain DEC VAX gear, and one program even requires propping up an ancient PDP-11, which the Smithsonian is interested in when it is finally retired.

"This is a situation where the commercial world would throw it out and move forward, but in the world of military contracts ten years is the minimum support requirement," she says.

Goldstone adds that a major problem felt across technology industries, the shortage of qualified talent to fill programming and engineering positions, is also impacting DCHS. It's her feeling that the booming Internet economy and the "skys the limit" salary structure in the Web world is also pushing salaries artificially high, and making it difficult for the less sexy realm of embedded systems to attract top talent.

"We're struggling to bring people into the embedded systems field, which is a big reason behind our desire for more efficient, automated, integrated programming tools. In the end its all cost driven, and everybody in the embedded systems industry is struggling with the same issue," she says.

Goldstone also sees the consumer markets nearly insatiable desire for new electronic devices like mobile phones and PDAs as a driving force for change in the safety-critical real-time embedded systems field, as the accelerating rate of change in the consumer sector forces organizations like DCHS to use increasing numbers of commercial parts, which have a tendency to become obsolete more quickly.

"Fortunately, commercial parts aren't a problem for us, since we anticipated the trend and significantly improved our screening processes in the early 1990s," she says. "Today, evaluating them is just another part of our development process, and all of our hardware designs pass through prediction tools that we use to help track how long specific parts should remain available."

While development tools like TADS are firmly ensconced at DCHS due to contractual obligations for projects like

the Apache flight control systems, another trend impacting their choice of future development environments is the movement toward commercial DSP chips in lieu of traditional processors.

"I think our sector of the embedded systems industry is increasingly moving toward miniaturization, just like everyone else," she adds. "They all want to find a way to generate more compact and efficient designs; basically everyone is saying, how many more parts can we get off the board and on this chip?"

She adds that DCHS is increasingly using C/C++ in their newer safety-critical system designs using DSPs for their speed, lower cost, and smaller size, and that she is currently struggling with the fact that the manufacturer of their DSP of choice the Intel TMS series isn't allowing tool vendors like DDC-I the access required to create programming tools for real-time safety-critical applications.

"Intel's limit on access is forcing us to use programming tools that are awful, some of the worst I've seen in a long time," she says.

She feels that integrated development environments capable of increasing the efficiency of both the individual engineer and the team are becoming essential. She specifically cites automated test generation tools, requirements generation that "flows," and a fully integrated toolset as required elements.

Goldstone also agrees with DDC-I's "partnership" approach to tools development: "I think they will be able to team up with hardware vendors using the DSPs and RISC-based processors to streamline the run time systems, since the embedded world doesn't use ninety percent of most kernels due to testing criticality."

In the end, she sees DDC-I's efforts with TADS positively. "From a customer support perspective DDC-I has been phenomenal. We had them create a support tool for us to handle unit test on a stand-alone board and they struggled incredibly to make it happen. I feel that they learned a lot [in the process] and provided a lot more depth in their group as a result," she adds. [...]

## USA / Lockheed Martin - Mark 41 Vertical Launch System

[From: "DDC-I Online News, July 2001, Vol. 2 Issue 5. -- dc]  
URL: [http://www.ddci.com/news\\_vol2num5.shtml](http://www.ddci.com/news_vol2num5.shtml)

Mark 41 Vertical Launch System Depends On DACS

Lockheed Martin leans on DDC-I's DACS-80x86 development tools to minimize maintenance costs for the

safety-critical core of the world-class Mark 41 Vertical Launch System

[...] Over the last twenty years, the battle-proven Lockheed Martin Mark 41 Vertical Launch System (MK 41 VLS) has revolutionized naval warfare, expanding the mission capability of surface combatants via its unique level of simultaneous multi-mission capacity. Delivering the widest range of ballistic ordnance available from a shipboard system, from SeaSparrows to Tomahawk cruise missiles, the system combines unmatched flexibility and unflinching performance on military ships stationed around the globe.

The U.S. Navy deploys the MK 41 VLS on AEGIS-equipped Ticonderoga-class cruisers and Spruance and Arleigh Burke-class destroyers, with plans to use the launcher on the next generation of surface ships. Onboard warships flying the flags of eleven different nations, the MK 41 launcher is deployed in thirteen configurations, ranging from a single eight-cell module to sixteen integrated modules totaling 128 cells.

The MK 41 launcher has been, or is currently being, integrated with sixteen different ship classes and eleven different weapon control systems. Capable of rapidly firing an aggregate of anti-aircraft, anti-surface, anti-submarine and land strike missiles, as well as theater ballistic missile defense munitions, the MK 41's primary mission is the quick delivery of deadly accurate ordnance during armed combat.

Safety-critical embedded systems where system failure is measured in human lives originated in the defense industry, and nowhere do they remain more important than on the modern electronic battlefield. Most important to the engineers keeping the launching system on the forefront of missile technology is the reliability of the real-time embedded systems at the core of the MK 41 VLS fire controls.

"We take safety-critical systems extremely seriously, we do layers and layers of tests," explains Staff Engineer David Farlow, the Integrated Product Team Lead for every deployed variant of the MK 41 launch sequencer still afloat. Farlow is based at the Baltimore, Maryland headquarters of the Marine Systems division, one of five components that comprise the Naval Electronics & Surveillance Systems (NE&SS) arm of the Lockheed Martin Corporation.

According to Farlow, safely firing a missile using the MK 41 VLS requires error-free communication between three primary components: The Launch Sequencer, Launch Control Unit (LCU), and the man-machine interface, the Weapon Control System (WCS).

When a launch order is given, the WCS fires a signal to one of two parallel LCUs

in each eight-cell launcher module, which issues prelaunch and launch commands for the selected missile. During normal operation one LCU controls the Launch Sequencer the critical communication link between the upstream fire control systems and the missile itself allowing the module to fire armament in tandem from two individual cells. Either LCU can control all eight cells if the other suffers damage.

"When you're firing a live missile, the first concern is for the safety of the people using the embedded systems that relay human commands to the launcher. Within our group, a fundamental part of system development is a strict process as part of code design and testing that scrutinizes specifically from a safety standpoint. We also have a whole separate group whose entire job is evaluating and prioritizing safety-critical development issues," says Farlow.

In addition to uncompromising safety, minimizing ongoing system costs is an important aspect of his role and responsibilities. Since the launching system has been in production since 1982, the embedded systems underlying the Launch Sequencer, LCU and WCS are well into the maintenance phase.

Fighting to maximize safety-critical performance and minimize maintenance and support costs for the MK 41 launcher, directly alongside Farlow and his team of engineers, is software development tools provider DDC-I. Their DACS-80x86 Ada compiler and tools maintain the assembly code for the run-time systems they embed on Intel 186 and 386 targets, which reside on separate boards within the Launch Sequencer.

DDC-I is a long-standing developer of Ada and C development systems and tools, including compilers, run time systems, and debuggers. Beyond their software engineering products, DDC-I also offers consulting and support services to a distinguished list of civil, commercial, and defense contractors, including Boeing, Sikorsky, and United Technologies.

The high level of importance assigned to controlling maintenance costs at Lockheed Martin is no mystery to DDC-I's Vice President of Technology, Joyce Tokar. "When you look at computer systems from the total lifecycle perspective, between sixty and eighty percent of costs occur after development and implementation. In other words, the maintenance phase accounts for the lions share of the cost in traditional projects," she details.

Like Tokar, Farlow has worked in the safety-critical embedded systems arena for well over a decade, and he feels that the largest improvements in software development have come at the hands of



vendors like DDC-I, who have dramatically improved the programming suites and testing tools available in recent years.

"I started working with embedded systems back in the late eighties, and the tools were very crude, black and white text-only displays. GUI environments have made it a lot more efficient. The biggest problem I'm running into now is finding modern tools that support older targets like the 186 and the 386," he states.

The best measure of DDC-I's steady focus on their customers needs is their continuing provision and support for programming tools that support older Intel processor cores like those embedded within the MK 41 VLS Launch Sequencer.

"We have broad experience with the bare-metal boys in the safety-critical embedded systems field, and our focus remains on getting the best possible application performance out of each processor," Tokar adds. [...]

## Indirect Information on Ada Usage

[Extracts from job-ads and other postings illustrating Ada usage around the world. -- dc]

*From: "Jim Harner"*

*<jlharner@home.com>*

*Date: Tue, 15 May 2001 22:52:30 -0700*

*Subject: Employment Opportunity*

*Newsgroups: comp.lang.ada*

I am a recruiter in Orange County, CA. I am currently doing an assignment for a company using Ada programming language on all of its current projects. I am looking for three or four highly trained Ada professionals who would be interested in joining a closely knit technical team in designing and developing sophisticated safety critical systems for use in the rail and transit industries. Specific requirements of these positions can be found in the attachment to this e-mail.

What is not reflected in the attachment is the terrific opportunity that these positions offer to an individual to work as part of a design team motivated to demonstrate the virtue of Ada and to increase train and rail safety far beyond levels which exist today. [...]

[Extracted from the attachment: -- dc]

Major division of leading railway electronics corporation. Located in Southern California about 40 miles east of Los Angeles, the division designs, manufactures and markets safety critical wayside and crossing signal products for the railroad and metro transit industries. [...]

Title: Senior Design Engineer.

Requires a minimum of 10 years experience in real-time embedded systems design. Must have strong background in Motorola and/or Intel microprocessors and expert skills in the use of Ada, the programming language being used on all projects currently in development. Person will have product life-cycle experience including requirements specification, system design, software architecture design, software design, coding/testing and integration. Should possess a broad knowledge of hardware, including software onto hardware integration and hardware debugging. Experience with railway electronics and/or safety critical systems is highly desirable. Must have BSEE, BSCE, or equivalent.

Title: Design Engineer

Requires a minimum of 4 years experience in developing real-time embedded systems. Must be able to do software design (partitioning software, drawing structure charts, etc.), coding and testing. Will have strong Ada programming skills, plus experience with Motorola and/or Intel microprocessors. Familiarity and knowledge of software integration is a plus, as is the ability to read hardware schematics and perform hardware debugging procedures. Must have BSEE, BSCE, or equivalent. [...]

*From: ganesh*

*<gkota@us.mascotssystems.com>*

*Date: Wed, 30 May 2001 18:09:14 +0200*

*Organization: Mascot Systems Limited*

*Subject: Looking for Experienced*

*programmers for Netherlands*

*Newsgroups: fido7.mo.job,comp.lang.ada*

[...] We are looking for Dynamic information technology professionals with good communication skills in written and spoken English for Mascot Netherlands operations. [...]

Ada Software developers with knowledge of embedded systems environment. We need specialists with experience in on-board software development for the satellite robot arm (or related technologies) and use of embedded processors like ERC 32 or TSC695 (Sparc alike), /Unix (Solaris), Ada (Verdix, Alsys), Emacs, LaTeX,. Experience in Simulation Technologies, Mathematical Modelling and Algorithmic programming is preferred.

Responsibilities: You will be involved in multiple phases of software development, including:

- embedded real time application development
- object oriented analysis and design methodology
- development of software standards within the software development lifecycle
- validation and verification testing

- exposure to Project Management using the CMM standard
- hardware and software integration and testing

Mandatory Skills: Ada and Assembler programming, university degree or equivalent background.

Assets: excellent communication and interpersonal skills, knowledge of software development practices & methodologies, and software documentation, experience developing and maintaining large mission critical systems.

*From: Anh Vo <anh\_vo@UDLP.COM>*

*Date: Thu, 31 May 2001 15:29:30 -0400*

*Subject: Re: Ada job wanted*

*To: team-ada@acm.org*

Lockheed-Martin is perfect for you. They are currently working a project called Space Based Infrared System (SBIRS) which using Ada95. Visit their website for more details. Good luck.

*From: Albert Galli*

*<albert\_galli@hotmail.com>*

*Date: Sun, 03 Jun 2001 15:32:26 -0400*

*Subject: Ada Developers*

*Newsgroups: comp.lang.ada*

Embedded Ada developers for exciting project in Toronto, Canada. You will aid in the development of navigation software for the next generation commercial aircraft. [...]

*From: "Webb, Geoff"*

*<Geoff.Webb@xwave.com>*

*Date: Fri, 8 Jun 2001 11:36:11 -0400*

*Subject: Xwave Employment Opportunity*

[...] an Embedded Developer opportunity utilizing [] Embedded development with either Ada or Pascal skills located in Toronto. [...]

xwave is an IT services company focused on clients in industries where we have extensive experience including energy, telecommunications, and select areas of the public sector. [...]

At xwave we asked ourselves whether we want to follow the crowd or walk our own path. The answer was obvious. Marching to the beat of our own drummer has helped us develop some of the most innovative solutions in the High Tech Industry like co-integration(tm), a revolutionary concept allowing us to combine realtime and mission critical operational systems together with corporate systems. It has also propelled us into a leadership position in the development of solutions in the areas of eBusiness, Aviation, Defense, Public Safety, Professional Services, and Advanced Systems. [...]

*From: wil@informate.be (Jean Williams)*

*Date: 19 Jun 2001 07:44:55 -0700*

*Subject: Help with Ada*

*Newsgroups: comp.lang.ada*



We are a OO software house based in Belgium, Europe.

For a project here in Brussels we need additional assistance from a person who has solid Ada experience. Any experience of developing safety critical solutions would be beneficial as would experience of UML, Teamwork, Rational Apex and IPSYS.

The project will last approximately 12 months. Due to the nature of the work, I'm afraid we can only accept European Union nationals. [...]

*From: "pserange Patrice Serange"*  
<pserange@alten.com>

*Date: Fri, 20 Jul 2001 12:07:38 +0200*  
*Subject: Fwd: New job opening*

Involved in embedded software engineers recruitment, among which several Ada specialists and engineers, [...]

Ada Software Engineer [Belgium]

Within our embedded competence center, we are looking for engineers and project managers specialized in Ada design and development. Projects are varied and could reach managerial responsibilities depending on your experience. The sectors involved are: aeronautic (design of avionic equipment) or railway. The evolution of this role can be international depending on your motivation. The knowledge of real-time constrained methodologies of development are an advantage, as well as experience in aeronautic or military standards. [...]

*From: Chris Kemper*

<chrisk@globalsourceit.com>  
*Date: Mon, 30 Jul 2001 17:14:02 -0500*  
*Subject: Job posting [...]*  
*To: team-ada@acm.org*

Looking for two or 3 junior level unit/integration testers for an opportunity in Belgium. The project should run August through April of 2002. [...]

*From: kepps@technisource.com*  
*Date: Mon, 13 Aug 2001 10:45:18 -0500*  
*Subject: Who Wants to be a Millionaire?*  
*To: team-ada@acm.org*

It's me again the mighty Technisource recruiter from Huntsville, AL. [...] Who can do Ada 95 Development, has experience with the Rational Suite Tools, a Secret Clearance, and is willing to contract in the "Rocket City" (Huntsville, AL)? [...]

Ada Development 95, Rational Apex, C++, Secret Security Clearance

*From: Jeff White*  
<jeffw@baldwinstaffing.com>  
*Date: Tue, 21 Aug 2001 14:21:16 -0600*  
*Subject: Ada Opportunities*  
*To: team-ada@acm.org*

My firm is currently representing a large corporation who has been awarded a military contract. They are looking for individuals who have strong: C/C++,

Rational Rose (Object Oriented design tool), Unix and Ada skills. [...] The opportunity would be based in Calgary, Alberta, Canada and covers all levels of skills from Junior to very Senior. [...]

*From: GeoLogics Corporation*  
<jbarry@geologics.com>  
*Date: Wed, 22 Aug 2001 14:07:04 -0500*  
*Organization: GeoLogics*  
*Subject: US-VA-TysonsCorner: Secret/Ada/XWindows/HPUX*  
*Newsgroups: comp.lang.ada*

US-VA-TysonsCorner: [...]. Consultant Support for the CCS Mk 2 Software Development Task

The following qualifications are necessary (in order of importance):

1. A current Secret security clearance is required
2. X-windows experience
3. Ada programming experience
4. HP UNIX
5. Creating a software architecture
6. Database/middleware experience (use, not design or development)
7. Mentoring programmers
8. 10+ years experience in the software development field
9. Familiarity with APEX DUO Integrated Development Environment
10. Pleasant, affable professional

[...] 6 month Renewable contract in Tysons Corner, VA, [...]

*From: David Horne*  
<dhorne@EXDIR.COM>  
*Date: Thu, 23 Aug 2001 14:58:46 -0700*  
*Subject: Senior Software Engineer in San Diego*  
*To: SIGADA-ABWG@acm.org*

I am a technical recruiter in the San Diego area. Our client is a leading provider of advanced information technologies, products, systems and services. Headquartered in the US, [...] Areas of expertise include command, control, communications, computers, intelligence, surveillance & reconnaissance (C4ISR); enterprise IT hardware & software solutions; information systems; training & simulation; science & technology; base & range support; and commercial information services.

Our client is looking for a number of Software Engineers, including Senior level engineers. This a permanent opportunity.

Description of Position:

Looking for experienced software engineers for new development project. Ideal candidate will have come out of a DoD environment. A secret clearance a plus but not mandatory. Must have experience with Ada and C++ and

tactical data link. This division is involved with tactical data links (TADILS), command and control procession (C2P), Multifunctional information distribution systems, and common data link management system (CDLMS). [...]

## Jobserve.co.uk Shows Positive Trend for Ada Jobs in UK

*From: "Martin Dowie"*  
<martin.dowie@baesystems.com>  
*Date: Fri, 6 Jul 2001 13:12:41 +0100*  
*Subject: Re: is ada dead?*  
*Newsgroups: comp.lang.ada*

I've been doing a little monitoring of jobs on www.jobserve.co.uk and Ada is the only language in my sample showing any signs of \_increased\_ use of the last 6 weeks!

Java down ~25%  
C++ down ~17%  
COBOL down ~10%  
C# static  
Pascal static  
Ada95 up ~33%

This follows the trend spotted by Computer Weekly in their quarterly job review (Ada up by 40+%/40+%/25%+ in the last 3 quarters - and the only language to show 3 consecutive quarters of growth). [See also next news item. -- dc] Obviously C++/Java are starting from a much larger base but COBOL is now only twice as popular as Ada (in the job advert stake anyway).

I haven't done any "proper" analysis - just counted the ads, but my assumptions are as invalid for one language as they are for any other, so I would hope that the trend is really following what is happening out there in the UK adverts! :-)

If anyone is interested I can keep the newsgroup posted as to how this sampling is going. Perhaps the compiler vendors could chip in with indications of how they see the market going? ;-)

As for defence only, it was recently reported (CW again?) that 1/3 of new Ada jobs are in telecoms...

## Computer Weekly Mentions a Continued Ada Revival

*From: "Martin Dowie"*  
<martin.dowie@baesystems.com>  
*Date: Mon, 20 Aug 2001 11:51:43 +0100*  
*Subject: Jobs survey*  
*Newsgroups: comp.lang.ada*

Taken from the latest Computer Weekly/SSP quarterly jobs survey:

"One of these is Ada, which has been enjoying a revival since 1998, when it

fell outside the top 60. This quarter it features in the top 20 for the third quarter in succession."

That's good news for Ada from this survey for the last year if I remember correctly.

[See also "Computer Weekly Survey: Ada Does Well Again" in AUJ 22.2 (June 2001), p.88. -- dc]

Article entitled "Is there a drought ahead?" in section "Homepage > Strategy > HR & Skills > Analysis" [at] www.cw360.com

*From: Dirk Craeynest*  
*Date: Tue, 21 Aug 2001*  
*Subject: Jobs survey (fwd)*

[From a discussion about the above mentioned article: -- dc]

Main downward trends: Java, HTML, C++, Corba, XML, etc.

Another quote: "It is the IT industry which is leading this return to more traditional development methods. [...] But users too have less enthusiasm for Java than they did."

> I'm surprised to see Corba going down. Considering how popular it's traditionally been, I didn't even know there /was/ a "down".

The quoted article in Computer Weekly mentions "under 150 job offers" for Corba in the last quarter vs. "more than 400" during the same period one year ago.

For comparison:

- Demand for "C++ skills": "fallen by more than 50%".

- Demand for "Java specialists": from "more than 2620" to "under 930", or a reduction by 64,5%.

For those who take the information from that kind of publications as gospel, seeing these downward trends reported there (too) should act as an eye-opener...

> Sorry to be a spoilsport here, but aren't these statistics of rather limited interest. All we see here is a trend in job ads. A lot of factors could influence these numbers aside from whether the particular language choice is so bad that it can cause projects to fail and ultimately the company to go bust.

I agree 100%. But reread the last paragraph above.

The interest of that kind of statistics is that it can be used to counter FUD such as "Ada is dead because there are so few job offers mentioning it".

For those people, the steady trends mentioned in the article should be revealing...

## On Programming Language Trends

*From: dewar@gnat.com (Robert Dewar)*  
*Date: 24 Aug 2001 09:59:06 -0700*  
*Subject: Re: Commentary on Industry Move to C++*  
*Newsgroups: comp.lang.ada*

[About moving to C++: -- dc]

> I don't know what is your industry. But industry in general has already moved to Java and VB and will be moving pretty soon to C#. According to recent statistics (don't remember the source - Gartner?... ) number of programmers using Java is larger than the number of programmers using C++

And the number of programmers using WORD macros is no doubt even higher, as is the number of programmers using Visual Basic and COBOL, but raw numbers like this don't tell you much.

Certainly in large segments of the industry, in particular the segments in which Ada is a large player, Java has only a slight presence, and is unlikely to make substantial inroads in the near future. It will be a while before we have either a commercial plane or a military fighter, or an air traffic control system entirely programmed in Java, if ever. As for the C# comment, this is mere speculation -- a lot of observers, even those who think that .NET will be wildly successful (itself not a forgone conclusion) think that other languages will dominate C# in this environment.

Just because Microsoft champions something does not mean it will succeed. [...]

## Ada in Context

### Standardization and Standard Conformance

*From: Florian Weimer*  
*<fw@deneb.enyo.de>*  
*Date: 24 May 2001 00:02:28 +0200*  
*Subject: Re: Do you have Standards Committee in your language?*  
*Newsgroups: comp.lang.ada*

> What is the role of Your Language committee?

[...] The Ada Rapporteur Group (ARG) is probably an institution which comes close (at least in function). It handles defect reports for the ISO Ada standard and examines future directions of the language (for the next revision of the standard, see <http://www.ada-auth.org/>).

> Does Your Committees expand horizons and enlightens compiler developers with their recommendations?

The compiler developers participate, of course. After all, they're quite familiar with the language, so their input is appreciated.

> Do you think that compiler vendors and users themselves are not able to maintain backward compatibility without such committees?

Wearing my Ada hat, I do not understand this question. However, if I put it off and try my C hat, I can understand what you mean. :-/

Ada vendors usually don't deviate from the language standard in any significant way, that's why we have the standard. ;-)

Some vendors provide additional features, of course, some of them are even comparable and compatible, but these extensions don't play a major role, unlike the 'extensions' over the standard many C implementations provide (mainly adherence to mostly unwritten conventions).

In contrast to many other popular programming languages, there is a publicly available test suite for Ada implementations, and compiler vendors seek independent validation of their products against this test suite because it's quite difficult to sell a compiler which hasn't been validated on at least some platforms. There's no market for proprietary Ada dialects at the moment, which is probably a good thing.

> And in general, do you see some analogies between Language Committees and other regulations in other areas of real life, on the net etc ?

IMHO, a group like the ARG is necessary if you have such a complex technical specification like a programming language standard. This is not a question of regulation, it's a technical necessity. [...]

*From: Tucker Taft <stt@averstar.com>*  
*Date: Thu, 24 May 2001 16:19:35 -0400*  
*Organization: AverStar (formerly Intermetrics), Burlington, MA, USA*  
*Subject: Re: Do you have Standards Committee in your language?*  
*Newsgroups: comp.lang.ada*

[On the ARG: -- dc]

ISO has a working group (working group 9, "WG9") devoted to the Ada language. This is the "language committee" in the official sense. The ARG is a subgroup of WG9. There are other "rapporteur" groups within WG9, for example the ASIS-RG, which works on standards related to Ada.

The ARG focuses on the language standard itself. The ARG makes recommendations to WG9. WG9 is the official body which takes the official country-by-country votes on these recommendations. They are then forwarded to upper level ISO groups

(JTC1, etc.) for further even-more-official action. [...]

All of our activities are quite open. The [<http://www.ada-auth.org/>] site mentioned above is the best place to look for minutes, technical reports, Ada Interpretations (AIs), etc.

[On compiler vendors validating their products: -- dc]

I agree with all of the above. Note that there are test suites for other languages which are publically available, but the Ada test suite (officially called the "Ada Conformity Assessment Test Suite" -- ACATS) has an official status with respect to the ISO standardized Ada conformance testing process. Also, the test suite is under continuous maintenance by the ACAA (Ada Conformity Assessment Authority), to ensure that it always reflects the latest rulings of WG9. [...]

Although in the beginning, the Ada test suite was a royal pain for implementors, it is now clear that to achieve true standardization, a rigorous test suite that is widely available is essential. My own view is that part of the failure of Java on the client was the inadequacy and lack of easy availability of a rigorous test suite for Java (especially for the GUI/AWT libraries).

Sun is still pretty stingy with their test suites, though we are at least now seeing more visibility of "J2EE certification." This seems like one of the few test suites that has achieved the level of visibility to the average users that the Ada test suite has had in the Ada community.

Tucker Taft, [stt@avercom.net](mailto:stt@avercom.net), <http://www.averstar.com/~stt/>, Chief Technology Officer, AverCom Corporation (A Titan Company), Burlington, MA, USA (AverCom was formerly the Commercial Division of AverStar)

*From: "Marin David Condic"*  
*<marin.condic@pacemicro.com>*  
*Date: Thu, 24 May 2001 11:17:36 -0400*  
*Subject: Re: Do you have Standards Committee in your language?*  
*Newsgroups: comp.lang.ada*

[...] Since Ada is one of the few OO languages out there with an ISO standard and test suite and a history of standard-adherence among compiler vendors, it would be interesting to see how other languages stack up against it. Other languages have standards, to be sure, but divergence from the standards seems to be more the rule than the exception. In this respect, Ada has done famously well - if one considers standard-adherence to be a good thing.

*From: "Marc A. Criley"*  
*<mcqada@earthlink.net>*  
*Date: Thu, 24 May 2001 17:35:41 GMT*  
*Organization: Quadrus Corporation*

*Subject: Re: Do you have Standards Committee in your language?*  
*Newsgroups: comp.lang.ada*

Actually there are language test suites for other popular languages and a testing center available for performing formal certifications (<http://eds-conform.com/ValProdList.html>). EDS provides conformance testing for Ada, C, C++, COBOL, Fortran, and SQL.

As Phil Brashear has periodically lamented, no one outside the Ada community has undertaken a language conformance assessment.

## Language Support for Fixed Point Arithmetic

*From: "Jean-Pierre Rosen"*  
*<rosen@adalog.fr>*  
*Date: Thu, 31 May 2001 11:01:54 +0200*  
*Organization: Adalog*  
*Subject: Re: Beginner's Language?*  
*Newsgroups: comp.lang.ada,*  
*comp.lang.lisp, comp.lang.smalltalk,*

[In reply to claims that having fixed point numbers in a programming language is very useful, for example for financial operations, Johan Kullstam <kullstam@ne.mediaone.net> replied: -- dc]

> You can always do fixed point with integers. For example, to do dollars to two decimal places, just count pennies instead of dollars. Other than a little notational convenience, i've never understood what the big deal was with fixed point.

Everything looks simple to people who didn't try to implement it...

Addition and subtraction are easy, but multiplication and division require some care. If you implement fixed points with integers, you'll end up doing by hand exactly what the compiler does for you, except that the people who wrote the compiler were educated in computer arithmetic, and believe me, there are lots of pitfalls there. (Said by someone who did an implementation of fixed point arithmetic).

*From: "Jean-Pierre Rosen"*  
*<rosen@adalog.fr>*  
*Date: Thu, 31 May 2001 15:51:47 +0200*  
*Organization: Adalog*  
*Subject: Re: Beginner's Language?*  
*Newsgroups: comp.lang.ada,*  
*comp.lang.lisp, comp.lang.smalltalk,*

"Raymond Toy" <toy@rtp.ericsson.se>:

> Why is multiplication/division much harder? You pretend the numbers [are] integers, do the operation, and then figure out where the fixed point should be. That's not too hard. As a DSP guy working on fixed-point DSPs, we do this all the time.

Yes, you "just" have to figure out where the fixed point should be.

Note that the result has twice more digits after the point than the original operands. This means that there is some rounding involved, and of course if you want to meet the numerics annex requirements, this should be done right.

And did you consider the case where operands are of different fixed point types, with different smalls, and the expected result type is a third fixed point type, or even a floating point type? And of course, you should consider arbitrary smalls, not just powers of 2... That makes determining the correct rounding of the result more interesting.

Of course, it is doable. Compilers do it all the time. But why bother to reinvent the wheel, when the compiler provides a carefully tested implementation?

## Arithmetic Overflow vs. Wrap Around

*From: "Frank A. Adrian"*  
*<fadrian@uswest.net>*  
*Date: Wed, 30 May 2001 21:15:23 -0700*  
*Subject: Re: Beginner's Language?*  
*Newsgroups: comp.lang.ada,*  
*comp.lang.lisp, comp.lang.smalltalk,*

[...] P.S. Any language where you can add two positive numbers together and get a negative number is inherently broken. It is a sign of the language designer and compiler writer choosing their own convenience over yours.

*From: thornley@visi.com (David Thornley)*  
*Date: Thu, 31 May 2001 19:22:50 GMT*  
*Subject: Re: Beginner's Language?*  
*Newsgroups: comp.lang.ada,*  
*comp.lang.lisp, comp.lang.smalltalk,*

[In response to "I presume [Java] is not the only language with such a stupid feature?": -- dc]

It's fairly usual in C. The C standard leaves the case of arithmetic overflow in signed types undefined, which means that the result could be negative, imaginary, text, or armed with thermonuclear weapons for all the standards care. Most systems I've used it on will wrap around, so that adding positive numbers and overflowing the limits will give you a negative number.

Partly the choice of languages depends on such things as whether you wish to get the right answer almost immediately or the wrong one with even greater speed. I never want to have to trust intermediate results not to overflow, which is one of the many reasons I like Common Lisp (in which case such overflow yields the right answer at a loss in speed).

Other interesting behaviors are truncation on the left because somebody left a 9 off a variable definition in COBOL. I've seen that cost thousands of dollars (well, to be honest, the client being billed probably saw it as a savings of thousands

of dollars). I've heard of such behavior in PL/I, a language I have never had the pleasure of working in.

You can also avoid some floating-point errors in Common Lisp by using rational numbers instead of floating-point. ( $1/3$ ) evaluates to an honest  $1/3$  rather than a floating-point approximation. If  $(+ (/ 1 3) (/ 1 3) (/ 1 3))$  doesn't yield 1, your CL system has problems. Pity about transcendental functions that have rational values on only a few points of their range.

David H. Thornley, david@thornley.net  
http://www.thornley.net/~thornley/david/

From: "Frank A. Adrian"

<fadrian@uswest.net>

Date: Sat, 2 Jun 2001 08:46:38 -0700

Subject: Re: Beginner's Language?

Newsgroups: comp.lang.ada,

comp.lang.lisp, comp.lang.smalltalk,

[In the same subthread, Johan Kullstam <kullstam@ne.mediaone.net> wrote: -- dc]

> It depends upon what you mean by "number" and what you mean by "negative". Sometimes you really do want to work in the ring  $Z_{2^{32}}$ . E.g., I often represent angles in terms of  $2^{32}$  quanta per revolution. When the 32 bit "int" wraps, so does the angle -- exactly as intended. I agree that it is a pain to have no way to use integers  $Z$ , but are forced to choose among  $Z_{2^N}$  for a few ill-defined choices of  $N$ .

Yes, I was talking about the set of integers. Most languages call the type something based on the word integer, leading the user to assume that one is, in fact, working with an element from that set. Although you are correct that it is sometime desirable to work with elements from alternative sets, I would also think that this is not the norm and that it would be better to actually declare the variables whose values are taken from these sets as such and not simply let the user to stumble upon the fact willy-nilly. One could also make a case that one should not canonize the size of these elements, defined by accident of hardware, but instead provide a specific data-type for any integral sub-ring the user desires. But then, that would ask the language designers and compiler writers to actually sweat the details.

Worse is better may be a model of what is, but it doesn't have to be a model of what will be...

## Integer Multiplication in Standard C

From: Ian Wild

<ian@cfmu.eurocontrol.be>

Date: Thu, 07 Jun 2001 12:55:37 +0200

Subject: hard sums

[Referring to: -- dc]

> The C standard leaves the case of arithmetic overflow in signed types undefined, [...] Partly the choice of languages depends on such things as whether you wish to get the right answer almost immediately or the wrong one with even greater speed.

OK - here's the question:

I was recently faced with the problem of trying to get the right answers for some simple arithmetic in C. If no right answer existed, I wanted to be informed of that fact, rather than just have a made-up number instead.

For addition it's relatively easy:

```
int safely_add (int a, int b)
{
  if ((a < 0) != (b < 0))
  {
    /*
     * if the input signs differ the answer is
     * guaranteed to be representable
     */
    return (a + b);
  }

  if (((a < 0) && (INT_MIN - a) > b) ||
      ((a > 0) && (INT_MAX - a) < b))
  {
    /* out of range */
    do_error_handling ();
  }

  /* we're safe if we get here */
  return (a + b);
}
```

Subtraction is similar, as you'd expect. Oddly enough, division is simplest of all:

```
int safely_divide (int a, int b)
{
  if (b == 0 || (a == INT_MIN && b == -1))
  {
    /* I happen to know I'm in a
     * twos-complement universe */
    do_error_handling ();
  }

  return (a / b);
}
```

However, I could find \*no way\* in standard C to safely multiply two integers at anything approaching a reasonable speed. Any hints, anyone?

From: Ian Wild

<ian@cfmu.eurocontrol.be>

Date: Thu, 07 Jun 2001 14:13:07 +0200

Subject: Re: hard sums

To: hat@cfmu.eurocontrol.int

Andrew Hatley wrote:

> Oddly enough, almost all processors will flag an arithmetic overflow or underflow, allowing the running software to instantly detect and attempt to handle the problem.

Well, yes, I know. I had seriously considered something like

```
int safely_multiply (int a, int b)
{
  _asm
  {
    mov eax, _a; mov ebx, _b; mul ebx;
    mov ebx, edx; cdq; cmp edx, ebx;
    jne do_error_handling
  }
}
```

but that relies on gcc extensions :-)

> It takes willful stupidity to ignore that flag and instead force the coder to add dozens of arithmetic manipulations of his/her own just to regenerate its value.

"C is efficient because it lets you get close to the hardware"

## Ada Saves the Day Again: SETI

From: Ted Dennison

<dennison@telepath.com>

Date: Mon, 04 Jun 2001 14:27:00 GMT

Subject: Ada saves the day again: SETI

Newsgroups: comp.lang.ada

The following is a copy of a post I just placed in alt.sci.seti. This message is \*not\* crosposted there. I reposted it here because I thought it might be of interest to folks in c.l.a. as well, for fairly different reasons. For one thing, I think its a good example of how a sensibly-written Ada program can be much more robust than a C program with the same amount of effort. For another, its an example of an effective way to perform Ada evangelism: by writing a generally-useful Free program with it, and then giving the language credit when its inevitably due. Most of the folks reading this are \*not\* programmers. But they have now been exposed to a scenario where Ada clearly was the more reliable choice.

[See also "SETI@Home Service 2.1" earlier in this AUJ issue for more information on the project. -- dc]

BTW: The subject was "missing fields in a SETI@Home text data file".

> Oops I take that back - according to the info on Roelof's SetiSpy homepage it seems like it is very much intentional from S@H's side, a move to improve security against "wu piracy"...

Ahhh, you're right. The following excerpt from the latest SETI@Home tech news explains it:

```
--- We closed the security hole with the
side effect that several fields in the
user_info.sah are now blank or zero.
We realize that this is a problem for
some very cool third party add-ons and
are putting some of the fields back. ---
```

As an aside: I don't spend too much space here pimping my favorite language, but

this time I can't resist. This problem was noticed due to a reported bug in the SETI@Home Service, which is entirely written in Ada. Due to "the magic of Ada", I was able to find the source of the problem without ever touching a debugger. The automatically displayed error message told me the source file and line at which the problem occurred, and that was all I needed.

The basic bug in the service's web server was that I searched for a certain character in one of the lines of the file (a `), and never bothered to account for the possibility that it might not be there at all. If I had made this exact same mistake in C, a core dump (or "Dr. Watson") would almost surely have resulted. That would have terminated the service, and quite possibly the SETI@Home clients too. However, Ada allows me to account for the possibility of unexpected errors, so that I can report them and recover. Thus all the SETI@Home service did when this bug occurred was display a message detailing the source line of the error, and continue on like nothing had happened.

In short, what would have been a nearly impossible to track down crash bug in C, became an easily isolated "hiccup" with Ada.

## Low Level Programming in Ada and C

*From: Mark Eichin <eichin@thok.org>  
Date: 16 Jul 2001 23:31:46 -0400  
Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

I find RTEMS useful as an example/proof-of-concept of [low level programming in Ada] - a complete realtime executive that runs on bare (m68k) hardware...

[See a.o. "GNAT/RTEMS cross-compilers" in AUJ 21.4 (January 2001), pp.231-232. -- dc]

*From: Roger Racine  
<rracine@draper.com>  
Date: Tue, 17 Jul 2001 07:47:03 -0400  
Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

Siddhartha Ray wrote:

> Is there any low-level programming that cannot be done in Ada but can be done in C?

Interesting question. I have been forced to use C for my current project, and when I turned on the appropriate flags to enforce ANSI C I got many, many errors. We were able to get rid of most of them, but it is very annoying. Portability of C code is possibly less easy than for Ada, based on my experience.

*From: Roger Racine  
<rracine@draper.com>  
Date: Thu, 19 Jul 2001 08:12:37 -0400  
Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

> I am not generally predisposed to defend C, but it seems to me that all C compilation should be done, from the start, with the most restrictive compiler setting to be used for the project. Whether it is annoying to write code that avoids those diagnostics is another matter.

Good point. I inherited the project and the code. I was actually trying to point out that (in my opinion) it was painful to write the code correctly.

The difficulties have mostly been pointed out, but I will mention one annoying ANSI C restriction: bit field components must be of type "unsigned int". This causes lots of (using Ada terminology) unchecked conversions to be inserted.

I will also point out that there were many weeks of effort that went in to solving problems that would have been caught by Ada's rules (like array indexes out of bounds).

*From: David Brown  
<gnatlist@davidb.org>  
Date: Thu, 19 Jul 2001 07:53:23 -0700  
Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

One of the most frustrating things I've found about C is how every compiler has radically different notions of what is a warning or an error. I used one compiler that would print messages such as:

```
foo.c 57 Warnings, 3 Errors, and 0
Serious Errors
```

and consider the file to have compiled correctly.

So many are trying to support old "definitions" of C. Such constructs as:

```
a;
which is a valid variable declaration.
```

One thing I greatly appreciate about Ada is the level of definition that the language has. When I have a question about what is valid, I can turn to the RM95 and see. If the compiler does something different, it is a definite bug.

With ANSI C, I can read a spec that somewhat describes what the compiler should check. So much of what is needed to do low-level programming is more of convention rather than defined, though.

Good example: Define a record with fields that are tightly packed. pragma Pack works quite well in Ada. It is even invalid to put aliased fields inside of a packed record.

Now try with C. Most compilers have some kind of extension to specify packing on a structure. They are often quite willing to take the address of various misaligned members within the structure, only to get memory access problems in distant parts of the code.

*From: Stephen Schwarm  
<sschwarm@emc.com>  
Date: Tue, 17 Jul 2001 16:49:15 -0400  
Organization: EMC Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

> Is there any low-level programming that cannot be done in Ada but can be done in C?

I think the more interesting question is: Is there any low-level programming that can be done in Ada but cannot be done in C? I think the answer to that is yes.

[And then the first poster asked: -- dc]

> Could you please send me an example. I am getting around to convince some people into using Ada at my company.

You can use rep clauses to define hardware object very accurately including unallowed values using enumeration. The examples in section 13.5 of the Ada Reference Manual is a start.

The first Ada project I did was using it as a design language for some micro code for some old bit slice processors. The micro coders were mostly hardware people and really loved the fact that the representations were complete, accurate and easy for all to understand. One can not come close with C field defs.

*From: "steved94" <steved94@home.com>  
Date: Tue, 17 Jul 2001 15:06:47 -0700  
Subject: Re: Low Level programming in Ada and C  
To: "GNAT Discussion List"  
<gnatlist@lyris.seas.gwu.edu>*

> I think the more interesting question is: Is there any low-level programming that can be done in Ada but cannot be done in C?

I think the answer is "no" unless you consider portability, then the answer becomes yes.

Ada permits defining structures that map to specific bits using representation clauses. C has something similar called "bit fields" but the language does not specifically define how the fields will be laid out for a particular compiler/architecture. With a particular C compiler you can figure out how it maps bit fields to the machine and code to that compiler, but if you switch compilers (even on the same architecture) don't expect the bit fields to line up in the same order.

*From: "Jeff Creem"  
<jeff@thecreems.com>  
Date: Tue, 17 Jul 2001 22:07:22 -0400*

*Subject: Re: Low Level programming in Ada and C*

*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

From: "Siddhartha Ray"  
 <SiddharthaR@TucsonEmbedded.com>

> Then why do so many projects use C to do some of the low-level programming even in an Ada project.

Interesting question and the answer varies from place to place. On one of my more recent projects (under vxWorks), the custom board support package was done under C since it was a small number of lines of code where we could reuse 50-70% of some existing code...

The rest of the "low level" code (device drivers for some 1553 chips, A/D chips, custom hardware, ISRs, etc) was done in Ada. Also board to board communication on the backplane was written in Ada (since the C approach built into the BSP had such high overhead -- no fault of C of course, just the protocol involved but things people often make comparisons of Ada implementations that are really not equivalent to some other C implementation that is faster so it only seems fair to continue the tradition).

Also (this was Ada 83) the math libraries were all written in Ada since the ones built into vxWorks were too slow. (One routine was a package machine code insertion to get direct access to the sqrt instruction and via pragma inline it flows in the code just like multiplies and adds).

*From: Culos Alain <Alain.Culos@IAA.ie>*  
*Date: Wed, 18 Jul 2001 09:55:33 +0100*  
*Subject: Re: Low Level programming in Ada and C*

*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

I think the simple answer is 'habit' and flexibility.

So many things have been written in C for low level and mostly communications code that people have a hard time changing their customs, call it inertia.

Another point is that people like the apparent ease and flexibility that C provides and do not always pay attention to the price to pay for that (unmaintainability amongst other factors).

Yet another factor is the pool of developers available. Many managers would rather buy in expertise that's common, widespread, readily available, cheap up front cost and not train them to a more suitable environment. So they do with what they know best, with what they have and combine those things to try and reduce their up front budget for any project.

*From: "Gautier de Montmollin"*  
 <gdemont@hotmail.com>  
*Date: Wed, 18 Jul 2001 14:53:46 -0000*

*Subject: Re: Low Level programming in Ada and C*

*To: "GNAT Discussion List"*  
 <gnatlist@lyris.seas.gwu.edu>

Maybe because C is a macro assembler that has allowed to develop small pieces of code (during the 1980s) that surely are used today, quite debugged. It is the same reason Fortran has still a dominant position in numerics & engineering: there are a lot of established code, although it is pain to develop new one (compared to Ada).

There is also a psychological effect: it is difficult to imagine that a verbose piece of Ada can produce a compact and fast machine code - although a "gcc -S -O2 -gnatp xyz" proves it...

## Generics in Java

*From: Terry Westley*  
 <twestley@buffalo.veridian.com>  
*Date: Wed, 6 Jun 2001 13:56:06 -0400*  
*Subject: generics in Java*  
 To: team-ada@acm.org

Some people should read the literature for design of other languages. Or, perhaps they just need to escape the NIH (not invented here) syndrome...

From <<http://www.zdnet.com/zdnn/stories/news/0,4586,2769630,00.html>>

Generics is controversial and has been widely debated inside and outside of Sun. Gosling said the concept is so complicated that no two developers could agree on exactly how it should be implemented. He and Sun Chief Scientist Bill Joy "got as close to physical violence as we've gotten," he said, because Joy wanted to slip Java for two or three years until Sun figured out generics. However, developers converged on an answer about two years ago, and then Sun waited while the issue wended its way through the Java Community Process. Gosling said reliability is critical to Java because it raises developer productivity, allowing them to "spend less of their lives fixing broken applications."

*From: "W. Wesley Groleau x4923"*  
 <wwgrol@fw.rsc.raytheon.com>  
*Date: Wed, 6 Jun 2001 16:56:45 -0500*  
*Subject: Re: generics in Java*  
 To: team-ada@acm.org

> Generics is controversial and has been widely debated inside and outside of Sun. Gosling said .... Bill Joy ... wanted to slip Java for two or three years until Sun figured out generics.

More evidence that Java was hacked, not engineered. At least the process of "fixing" Java is better than the process that dumped it on the world.

*From: Tucker Taft*  
 <stt@mail.burl.averstar.com>  
*Date: Wed, 6 Jun 2001 17:37:35 -0400*  
*Subject: Re: generics in Java*

To: team-ada@acm.org

One point of interest: Norm Cohen, who has been involved in the Ada community for many years, both as a language lawyer and as a book author, is one of the co-authors on the April 2001 draft of the proposal to add generics to Java. That makes me feel it is more likely that the lessons learned in Ada will be taken into account.

## DII COE Prohibition of Ada to JBC Compilers Rescinded

*From: "Marc A. Criley"*  
 <marccriley@earthlink.net>  
*Date: Tue, 19 Jun 2001 07:51:05 -0400*  
*Subject: DII COE Prohibition of Ada (et.al.) to JBC compilers rescinded*  
 To: team-ada@acm.org

About a year ago I discovered that the DII COE [Defense Information Infrastructure Common Operating Environment -- dc] Integration and Run-Time Specification (I&RTS) 4.0 prohibited the use of tools that compiled languages other than Java into Java Byte Code (JBC). This obviously precluded the use of Ada products such as JGNAT and AppletMagic to develop "portable Web-based" applications. (The use of analogous compilers for other languages, such as Eiffel and Python, were therefore barred as well.)

Here's the original prohibition:

"Developers shall not use compilers designed to convert code developed in other languages (e.g., Ada, C++) to create Java byte-codes. This restriction is important because such compilers may inadvertently bypass intended Java security features." -- Section 8.2.3, bullet 3.

The original rationale for this in the I&RTS was an incorrect belief that the JBC resulting from the compilation of other languages could bypass Java's security features, and that only by using Java as the source of class files could adherence to the security features be ensured. This was a clear misunderstanding of Java's security model, as it is the responsibility of a Java Virtual Machine (JVM) to enforce security. To the JVM (and its security implementation), how a class file comes to exist is irrelevant.

An objection to this prohibition was created and discussed in this forum in April of 2000, and with a large number of co-signers (some from outside of the Ada community as well), was submitted to the DII COE Chief Engineer, in accordance with the process described in the I&RTS itself. [...], I went to see if an I&RTS update had been released--which had, it's now version 4.1. I looked up the section corresponding to the original

"Miscellaneous" section, which has been renumbered as 8.2.4.

I was pleased to find that there is no longer any mention (pro or con) of the means used to produce JBC class files. Although our original submission suggested alternative wording for this bullet, simply removing it can certainly be counted as removing an unjustified barrier to Ada.

[...] I have no idea what role our submission played in the decision to remove the bullet. Perhaps major, perhaps none, perhaps it was the Python advocates that got it out :-). Whatever the reason, we did our part, and I'm pleased with the way things turned out.

Marc A. Criley, Senior Staff Engineer,  
Quadrus Corporation,  
www.quadruscorp.com

P.S. A plaintext version of the submission is available in the Team Ada archives at  
<http://www.acm.org/archives/wa.cgi?A2=ind0004&L=team-ada&P=R2106>

## On Java vs. Ada

*From: James Rogers*  
*<jimmaureenrogers@worldnet.att.net>*  
*Date: Sat, 07 Jul 2001 18:33:12 GMT*  
*Subject: Re: is ada dead?*  
*Newsgroups: comp.lang.ada*

[In response to claims that Ada is "too old"... -- dc]

[...] I am amazed that the senior lecturer was unchallenged in his or her statement about Ada being too old. I expect this is in comparison to Java. This amounts to saying that a proven product is not as good as an unproven product because the unproven product is newer. Such a statement is pure nonsense.

It is even more nonsensical when you realize that Java has very few new language developments. It is simply a compilation of a lot of language features from a lot of other languages. In fact, without the standard Java API the language would be completely ignored. It offers nothing new, and it offers all these old features with very poor performance.

I would have been tempted to ask the senior lecturer if he or she was willing to fly in an airplane with avionics programmed in Java.

Look at Java's thread model. It is seriously flawed. This is not just my opinion. This is the opinion of the Java development team themselves. The earliest Java thread model included methods stop() and suspend(). After several years of use it was discovered that those methods are so unsafe that they had to be deprecated in the language. The use of each method would frequently result in deadlocks and race conditions. The major

problem was not that those methods were faulty, but that the thread model was designed such that those methods could not be properly fixed. The only option left to the Java design team was to simply declare those methods to be broken, and advise against their use.

Java tried one somewhat new idea for dealing with threads. That is the concept of thread groups. The idea was that threads could be associated with a group, allowing all the threads in a group to be stopped or suspended at the same time. Now that the stop() and suspend() methods are deprecated, there is no reason to use thread groups. Thus, the one attempt at real invention in the Java language was made redundant because the underlying thread model was so poorly designed.

Java was originally touted as the answer to client-side programming in a networked environment using applets. Applets are java applications running in a user's browser. For several years people tried to make applets work. Some success was achieved. Then Java 2 was released, with its Java Foundation Classes including the Swing gui components. Swing is supposed to be usable with applets as well as regular applications. Unfortunately, Netscape and Internet Explorer have chosen to ignore Sun's advice on HTML support for Swing applets. Each has chosen its own approach to supporting Swing, resulting in a most horrible nightmare of HTML code to merely start a Swing applet. Even worse, there is no assurance that Netscape and Internet Explorer will not change their own HTML syntax to support Swing. The result is that Swing is essentially unusable in applets. Most companies simply use dynamically created HTML and CGI to perform client-side programming. Microsoft has created Application Server Pages for this purpose. Sun followed up with Java Server Pages.

Applets are simply another feature of Java found to be of little practical use.

Java is an unstandardized language. Sun likes to call this a de-facto standard. Sun has its own meaning for such a phrase. It means that everyone is encouraged to use Java. The Java API documentation is freely published on the web (even though the API documentation contains some serious errors). Sun is free to change the API documentation at any time. The thread example above is a good example. Sun maintains complete control over what Java is and is not.

Sun has started into formal international standardization of Java no less than five times. Each time they have pulled out of the effort. It is not clear to me that Sun will ever cooperate with a standardization of Java.

Java is clearly not superior to Ada. Java is not even clearly newer than Ada. Java is clearly more unstable and unsafe to use than Ada.

*From: James Rogers*  
*<jimmaureenrogers@worldnet.att.net>*  
*Date: Sat, 07 Jul 2001 18:44:22 GMT*  
*Subject: Re: is ada dead?*  
*Newsgroups: comp.lang.ada*

Larry Kilgallen wrote:

> I don't think [the C, C++, Java, and C# family] will ever forsake their null-terminated strings.

That may be true for C, but C++ now has a String class, which is not just a wrapper for the more primitive C strings. Java has a String class which is also not null terminated. The Java String class is closer to an Ada Unbounded string, with everything dynamically allocated.

Of course, the Java String suffers from the same inefficiencies as an Ada Unbounded string. For instance, you cannot edit a Java String. There are Java methods to change the value of a character at a specified position. The result, however, is not strictly an edited string. It is an entirely new string, with all the characters copied and the edited character replacing the original. In other words, if you want to loop through all the characters in an existing 1024 character String, replacing each one, you will require the creation and garbage collection of 1024 1024 character strings. Since Java characters are all 16 bits, this means that you need to chew up over 2 Megabytes of data to edit a 1024 character String.

Clearly, Java has abandoned the C null terminated string. Clearly they have also abandoned any approximation of efficiency in String handling.

## Microsoft Windows XP Will Not Support Java

*From: "Robert C. Leif, Ph.D."*  
*<rleif@rleif.com>*  
*Date: Wed, 18 Jul 2001 23:05:28 -0700*  
*Subject: Good Bye Java*  
*Newsgroups: comp.lang.ada*

Evidently, Sun's control of Java gave Microsoft the opening it needed.

First there was C, then C++, then Java, and now maybe C#. We Ada people can watch and laugh. However, Java should have taught us the power of marketing.

[The rest of this message was an excerpt from <http://www.msnbc.com/news/601794.asp#BODY> -- dc]

Microsoft pulls back support for Java, in blow to rival technology

By John R. Wilke and Don Clark The Wall Street Journal

"Prerelease copies of Microsoft's new Windows XP operating system, which



goes on sale this fall, drop the software needed to run Java-based programs. [...] Java software is used to create some of the animated and interactive features of Web pages and hand-held devices; Web surfers using computers with Windows XP won't see those features without loading additional software."

[The full article is no longer available at the URL above, but <http://www.zdnet.com/zdnn/stories/news/0,4586,2790355,00.html?chkpt=zdhpnnews01> still has a copy.]

## Static Checker for Java Cannot Reach Ada's Strong Type Checking

From: *peteclose@-*

*<peteclose\_member@newsguy.com>*

Date: 16 Aug 2001 13:01:25 -0700

Subject: *is Java getting close to Ada strong type checking with this tool?*

Newsgroups: *comp.lang.ada*

<http://research.compaq.com/SRC/esc/>

"The Compaq Extended Static Checker for Java (ESC/Java), developed at the Compaq Systems Research Center (SRC), is a programming tool for finding errors in Java programs. ESC/Java detects, at compile time, common programming errors that ordinarily are not detected until run time, and sometimes not even then; for example, null dereference errors, array bounds errors, type cast errors, and race conditions.

ESC/Java uses program verification technology, but feels to a programmer more like a type checker. By using an underlying automatic theorem prover, ESC/Java is more semantically thorough than decidable static analysis techniques. At the same time, because it tries to detect certain kinds of errors only, and not prove the program's correctness, the technique is more automatic than full functional program verification."

From: *Tucker Taft <stt@avercom.net>*

Date: *Fri, 17 Aug 2001 05:55:41 -0400*

Organization: *AverStar*

Subject: *Re: is Java getting close to Ada strong type checking with this tool?*

Newsgroups: *comp.lang.ada*

The key difference is that programmers can't communicate as much in Java as they can in Ada. For example, all arrays in Java are indexed by integers 0..length-1. There are many applications where this is not the natural index type for the array. Arrays are essentially like functions, and the index type should be strongly typed, just like the parameters to a function. Not all functions are presumed to take an integer parameter ranging from 0..N-1!

Perhaps even more glaringly is that Java doesn't have enumeration types, nor any

user-defined scalar types (not even user-defined names for the predefined scalar types). That means that a parameter type is often simply "int" when in fact the only legal values are much more restricted, and correspond roughly to an enumeration. There really is no reasonable excuse for having omitted enumeration types from Java (I have read the "official" rationale about lack of extensibility and it is extremely weak in my view).

Unfortunately, there is no way for an enhanced static checker to overcome some of these limitations. There simply isn't enough information.

On the other hand, I am a very big fan of enhanced static checking, and I think it should be an important area of focus for programming language research.

From: *"Warren W. Gay VE3WWG"*

*<ve3wwg@home.com>*

Date: *Fri, 17 Aug 2001 18:17:24 GMT*

Subject: *Re: is Java getting close to Ada strong type checking with this tool?*

Newsgroups: *comp.lang.ada*

It was also pointed out in a paper I recently read, that all Java methods are dynamic (virtual). This also eliminates some compile time (static) checking that otherwise could have been performed.

## Java and Aerospace

From: *Gary Scott*

*<Gary.L.Scott@lmtas.lmco.com>*

Date: *Fri, 24 Aug 2001 08:53:00 -0500*

Organization: *LM Aeronautics*

Subject: *Re: Commentary on Industry Move to C++*

Newsgroups: *comp.lang.ada*

Andrzej Lewandowski wrote:

> I don't know what is your industry. But industry in general has already moved to Java and VB and will be moving pretty soon to C#.

Aerospace/embedded processors. There's absolutely no way anyone would consider Java given current processor performance and real-time processing requirements.

From: *James Rogers*

*<jimmaureenrogers@worldnet.att.net>*

Date: *Sat, 25 Aug 2001 01:51:52 GMT*

Subject: *Re: Commentary on Industry Move to C++*

Newsgroups: *comp.lang.ada*

Even more direct than that, look at the license agreement for the Java development kit. It specifically states that the user will not use the kit to produce software to control airplanes, or air traffic control systems.

I assume Sun has some reasons for insisting you know not to use Java for these purposes.

From: *James Rogers*

*<jimmaureenrogers@worldnet.att.net>*

Date: *Sat, 25 Aug 2001 07:12:50 GMT*

Subject: *Re: Commentary on Industry Move to C++*

Newsgroups: *comp.lang.ada*

Al Christians wrote:

> They say it's because of timing -- i.e. because of garbage collection -- they can't make it a sure thing that it will accomplish anything in a given amount of time. [...] That's really a JVM issue, not a Java issue -- there are many languages running on JVM these days.

Timing is a JVM issue. It is also a Java issue.

My experience is that the Java language in general, and Java standard API classes in particular, are relatively inefficient. Java static data members are allocated on a stack. All other Java non primitive data items are allocated dynamically.

Aside from the fact that additional overhead is involved in marking and deallocating dynamic memory, the most common Java programming models allocate a lot of memory when other languages like Ada, C, Basic, Pascal, etc will not need to allocate memory. This addiction to allocation produces performance problems that affect both timing and memory utilization.

For instance, using a Java collection, such as a HashMap to store values associated with unique keys requires that all items stored must be instances of classes, not primitive types. How, then, do you store numeric types in a HashMap? Simple. You convert the numeric type to its corresponding wrapper class.

The primary reason to use a HashMap is for rapid access to the values contained therein. If you want to total values read from an input stream by key value you will need to increment the values stored in the HashMap. Unfortunately, the numeric wrapper classes are incapable of performing math operations. You must convert the wrapper class object to a primitive type, perform the calculations, then create a new numeric wrapper object and put it in the HashMap under the same key. All this conversion, allocation, and reassignment is expensive. Timing is affected greatly.

(Of course, each of those objects must be eventually deallocated or "collected" by the GC, which even if it were deterministic would still have a serious affect on timing.)

Java also demonstrates plenty of unpredictability with its thread model. Releasing a thread from a "wait" call is accomplished by having another thread call the "notify" or the "notifyAll" method. The "notify" method causes the virtual machine to release one thread from the wait state. There is no sense of a queue in the "wait" state corresponding to



an Ada entry queue. You have no way of knowing which thread will be affected by a "notify" call. It may be one that must immediately return to the "wait" state because its boundary condition has not been satisfied. If so, then you can encounter a deadlock because the expected thread was not activated.

The common wisdom is that one should never use "notify". Instead, one should always use "notifyAll". "notifyAll" releases all threads from the "wait" state. This avoids deadlock, because one of the waiting threads will discover a satisfied boundary condition and continue. All other threads will have to return to the "wait" state. This approach avoids deadlocks but produces timing variations, depending on the number of threads in the "wait" state. Java provides no remedy for this timing issue.

## Porting from Ada to C++

*From: prabhath*  
*<prabhaths@shiparatech.com>*  
*Date: Thu, 7 Jun 2001 12:00:05 +0100*  
*Subject: Porting from Ada to C++*  
*To: team-ada@acm.org*

I have a requirement, where in I have to port the application written in Ada to C++. Is there any tool available for porting?

*From: Ben Brosgol*  
*<brosgol@GNAT.COM>*  
*Date: Thu, 7 Jun 2001 11:16:34 -0400*  
*Subject: Re: Porting from Ada to C++*  
*To: team-ada@acm.org*

This sounds like a technique, not a requirement, and there may be better approaches such as keeping the application in Ada and then interfacing with C++.

*From: Tucker Taft*  
*<stt@mail.burl.averstar.com>*  
*Date: Thu, 7 Jun 2001 09:31:11 -0400*  
*Subject: Re: Porting from Ada to C++*  
*To: team-ada@acm.org*

AverCom has a technology for translating Ada 95 to pure C, or to a subset of C++. This technology is available in two forms, one as an Ada 95 compiler, which just happens to use optimized ANSI C as its intermediate representation. This allows you to continue to maintain the Ada source, while using a C or C++ compiler for the final compilation process.

In the other form, a one-time translation is performed, carrying over the Ada comments and the Ada source code structure into the generated C/C++. Further development is then done by editing the generated C/C++. We have generally made this latter form available only as a service, with a fee based on number of lines of code to be translated.

Of course switching to using C or C++ as the source language involves significant

loss in compile-time checking on any further changes to the software. Keeping with Ada as the source language ensures that all of the compile-time and run-time checking provided by Ada is continued during further changes to the system. [...]

## Porting from Ada 83 to Ada 95

*From: prbhath s*  
*<prabhaths78@hotmail.com>*  
*Date: Mon, 25 Jun 2001 14:00:01 -0000*  
*Subject: porting from Ada 83 to Ada 95*  
*To: team-ada@acm.org*

[The same poster as above, but a few weeks later... -- dc]

I have a requirement where in I have to port code from VADS Ada 83 compiler to GNAT Ada 95 compiler. My query is, if there is any tool available for this work or any guideline sort of thing which helps in porting.

*From: "David C. Hoos"*  
*<david.c.hoos.sr@ADA95.COM>*  
*Date: Mon, 25 Jun 2001 10:25:34 -0500*  
*Subject: Re: porting from Ada 83 to Ada 95*  
*To: team-ada@acm.org*

Having done this a few times, I can speak with some degree of authority on the subject.

Ada83 syntax is something like 99.9% compatible with Ada95. The largest single compile-time error I encountered in this arena was the changed syntax for generic units with unconstrained actual subtypes. Here is that difference:

Ada95 provides new syntax for a generic formal private type to indicate that the actual subtype is allowed to be indefinite (i.e. unconstrained without defaults). The old syntax is retained, but the meaning is changed to require definite actuals (i.e. constrained or with defaults).

A full discussion of Ada83/Ada95 compatibility may be found at <http://www.adaic.org/docs/compat-guide/compat-guide6-0.txt>

However, the largest part of the effort is dealing with calls to compiler-specific libraries. Many features of Ada95 -- e.g., (the Ada.Command\_Line package) were unspecified in Ada83, so each compiler vendor was free to implement his own libraries -- and VADS in particular has a number of library units typically heavily-used that are not compatible with the Ada95 language-defined libraries.

Depending on your platform, the folks at gnat.com may have some libraries with the VADS interfaces that can be used as a compatibility layer.

My own approach for much of this was to replace the VADS bodies with implementations that invoke either the standard Ada95 libraries, and/or the

florist implementation of the POSIX/Ada95 bindings.

Depending on how well the designers of the original code isolated compiler-specific library calls, and how much code there is, the amount of work to be done in this arena can be significant. [...]

*From: Terry Westley*  
*<twestley@buffalo.veridian.com>*  
*Date: Wed, 27 Jun 2001 17:14:18 -0400*  
*Subject: Re: porting from Ada 83 to Ada 95*  
*To: team-ada@acm.org*

We recently completed converting ~750K SLOC of VADS Ada 83 code to GNAT Ada 95. I know of no tool to do the heavy lifting for you. Here are some tips that may prove useful. I concur with David Hoos that the major difficulty is in how much you used VADS-specific features (such as passive tasking) and VADS library calls.

First and foremost, do NOT ignore any GNAT warning messages unless you understand the warning and choose to leave it there. We tried to eliminate all warning messages by correcting the code. Trust the compiler; it's your best tool.

We didn't have the luxury to stop all development while doing the conversion, so we preserved VADS implementations in package bodies along with parallel Ada95 implementations for awhile.

1) Use gnatclop to separate compilation units into separate files and to rename files as needed.

2) We created our own packages which mirrored the purpose of some of the VADS packages. Each of these packages had two bodies, the VADS/Ada83 body and the GNAT/Ada95 body. As it turns out, very little code was GNAT-specific since there's lots more predefined packages to handle things like unbounded strings, command line arguments, environment variables, and random numbers, and exception information in Ada95 than in Ada83.

[More detailed suggestions deleted. Check the complete message in the team-ada archive at <http://www.acm.org/archives/wa.cgi?A2=ind0106&L=team-ada&F=&S=&P=5105> -- dc]

Terry Westley, Veridian Engineering,  
 twestley@buffalo.veridian.com

## Porting to the Current Hot New Language

*From: Ted Dennison*  
*<dennison@telepath.com>*  
*Date: Wed, 22 Aug 2001 19:35:03 GMT*  
*Subject: Re: How Ada could have prevented the Red Code distributed denial of service attack.*  
*Newsgroups: comp.lang.ada, comp.lang.c, comp.lang.c++, comp.lang.functional*

Jerry Petrey wrote:

> [...] We had a very good implementation of the engine controller in Ada but the management was so poor that they allowed it to be re-written (after I left) in C or C++ from what I've heard - to be more 'politically correct'. That was their downfall.

Well, bad management is everyone's downfall unfortunately. I think Jerry's referring to another R&D engine controller, [...]. The manager in charge of that was just about the worst kind you can have: the idiot who thinks he is a genius. An idiot manager who knows he's an idiot and sticks to leading and listening can actually be quite good, but this other kind just destroys everything he touches.

I can remember the IM (idiot manager) informing a visiting prospective customer that we were porting that perfectly working engine controller to C++ from Ada. When the customer incredulously asked why we'd do such a useless thing, IM told him essentially that he, the customer, would refuse to buy it no matter how good the specs, if it were coded in Ada internally rather than the current hot new language. I'm guessing IM truly believed this. Apparently the prospective customer was not horribly impressed with IM's sensitivity to his heretofore undiscovered coding language "hipness" desire for his engine controllers, because he never did buy anything from us. :-) [...]

## Official Recommendations of Ada

*From: 18k11tm001@sneakemail.com (Russ)*

*Date: 17 Jul 2001 00:13:04 -0700*

*Subject: Official recommendations of Ada*  
*Newsgroups: comp.lang.ada*

I work in an environment dominated by C/C++, and I would like to recommend Ada for a safety-critical application that is about to be initiated. I think official recommendations of respected standards organizations will carry a lot of weight. I came across a reference to IEC-1508, for example, which apparently recommends Ada. [...] [Does] anyone know of other official recommendations of Ada over other languages for critical software (either online or available through snail-mail)? Thanks.

*From: Kilgallen@eisner.decus.org (Larry Kilgallen)*

*Date: 17 Jul 2001 09:37:59 -0500*

*Organization: LJK Software*

*Subject: Official recommendations of Ada*  
*Newsgroups: comp.lang.ada*

Pat Rogers <progers@classwide.com> writes:

> The MISRA C guidelines recommend Ada (among others) at the end of the document. <http://www.misra.org.uk/>

Good pointer. Thanks (even though I did not ask the question).

In particular, the freely-distributable just so you do not excerpt preview at <http://www.misra.org.uk/graphics/miscprev.pdf> give that recommendation in the penultimate paragraph of page 3. The only better-than-C languages for safety-related systems they call out by name are Ada and Modula 2.

*From: hfrumblefoot@yahoo.com (Hambut)*  
*Date: 17 Jul 2001 14:32:51 -0700*

*Subject: Official recommendations of Ada*  
*Newsgroups: comp.lang.ada*

It's been a dull evening tonight, so I thought I'd have a wade through the stuff I've got to hand to see what I can find with respect to Ada recommendations. This may help - it might at least be a starting point.

I've managed to find three broad categories of things:

1. Recommendations to use Ada in safety related applications
2. Links to articles which perhaps provide ammunition against the use of C++.
3. Some general safety links that you might find useful if your company's about to embark on a safety related project.

1. Recommendations to use Ada in safety related applications

I've managed to find 3.5 concrete recommendations for the use of Ada in safety related systems:

- 1.1 European Railway Safety Standard prEN50128
- 1.2 MISRA C Coding Guidelines
- 1.3 US DoD Joint Software System Safety Handbook (the half a recommendation)
- 1.4 A paper published in the early nineties

Interestingly neither DO178B (aviation software safety standard) nor Defence Standard 00-55 (UK defence standard for the development of safety related/ critical software) make recommendations as to language choice. Instead they define characteristics that the chosen language should obey. The characteristics defined by Defence Standard 00-55 pretty much boil down to a choice of SPARK Ada [...] for the highest integrity levels.

I believe that IEC61508 (you referred to it as 1508 in your initial posting) has a similar table to prEN50128 - although I'd need to chase it up to be absolutely sure (so all you standards junkies don't shoot me down in flames just yet).

Final point to note is that there is a very useful document being produced by an ISO committee (The Annex H Rapporteur(sp?) Group) called "Programming Languages - Guide for the Use of the Ada Programming Language in High Integrity Systems". [...] [See below for more information. -- dc]

1.1 European Railway Standard prEN50128

Table A.15 of CENELEC prEN50128 lists Ada as being "highly recommended" for Safety Integrity Levels 1 and 2, and "recommended" for Safety Integrity Levels 3 and 4.

The same table lists unrestricted C or C++ as "not recommended" for safety integrity levels 3 and 4, and makes no recommendation for safety integrity levels 1 and 2. However it also lists "A subset of C or C++" as "recommended" for safety integrity levels 1 to 4.

Interestingly it doesn't mention a safe subset of Ada in the table. However it does say, in a note, "At Software Safety Integrity Level 3 and 4 when a subset of languages 1,2,3 and 4 are used the recommendation changes to HR" HR is the code for highly recommended. Ada is language 1. So when a suitable subset of Ada is used the recommendation is "highly recommended".

This is from "Railway applications - Software for railway control and protection systems", prEN 50128. Possibly available from [www.cenelec.org](http://www.cenelec.org).

1.2 MISRA C Coding Guidelines

"Examples of languages generally recognised to be more suitable than C [for safety related software] are Ada and Modula-2. If such languages could be available for a proposed system then their use should be seriously considered in preference to C."

Document at: - <http://www.misra.org.uk/graphics/miscprev.pdf>

1.3 US DoD Software Safety Handbook

"The Ada programming language provides considerable support for preventing many causes of unpredictable behaviour allowed in other languages. For example,....., implicit constraint checks prevent the classic "C" programming bug of writing a value into the 11th element of a 10-element array."

This is from "Software System Safety Handbook - A technical and Managerial Team Approach", Joint Software System Safety Committee, December 1999.

Available from [http://www.nswc.navy.mil/safety/joint\\_software\\_system\\_safety\\_han.htm](http://www.nswc.navy.mil/safety/joint_software_system_safety_han.htm)

1.4 The choice of computer languages for use in safety-critical systems

A paper published in the early nineties, which looked at a number of attributes that the authors thought were important for 'safe' software, has this to say about the subject:

"The languages that design teams should consider as candidates for use in high integrity systems are, according to the assessments in this paper, and in descending order of merit

- ISO Pascal subsets supported by validation tools (e.g. SPADE Pascal);
- an Ada sub-language, when available;
- a Modula-2 sub-language when available;
- a CORAL 66 subset."

"If analysis of the hazards suggests that the risks are comparatively low, the second group of languages that may be considered includes, in no particular order

- structured assembly languages;
- DoD Ada, with minimal restrictions;
- ISO Pascal, with minimal restrictions;
- Modula-2, with minimal restrictions."

"Based on the assessments in this paper, the use of the following languages is to be deprecated when safety is an issue

- unrestricted use of assembly languages;
- C (despite its many adherants)
- unrestricted use of CORAL 66"

This is from "The choice of computer languages for use in safety-critical systems", W.J.Cullyer, S.J. Goodenough and B.A. Wichmann, Software Engineering Journal, March 1991.

I believe this study is also quoted in Neil Storey's "Safety Critical Computer Systems" book.

2. Links to articles which provide ammunition against the use of C++

There are two main links.

For a critical view of C++, which may provide useful reasons why it should not be used in a safety related/critical project look at the Joyner paper, "A Critique of C++ and Programming and Language Trends of the 1990s" available at: <http://www.progsoc.uts.edu.au/~geldridg/cpp/cppcv3.html>

And for a debate, which includes submissions by a number of respected safety people about the use of C++ in developing safety related software take a look at: <http://www.cs.york.ac.uk/hise/sclist/cplussafety.html>

3. Some General Safety Links You Might Find Useful

In fact there's one, which is a good starting point for links to lots of other sites interested in software safety: <http://www.afm.sbu.ac.uk/safety/>

Hope this helps.

From: "Martin Dowie"

<[martin.dowie@baesystems.com](mailto:martin.dowie@baesystems.com)>  
Date: Wed, 18 Jul 2001 11:08:08 +0100  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

ARINC Report 613, Guidance for Using the Ada Programming Language in Avionic Systems

"It is the desire of the airline community to reduce the cost and the economic risk associated with avionics software systems. As a means to achieve this, it is recommended that the Ada programming language be used as the standard High-Order Language (HOL) in avionics equipment design."

From: [hfrumblefoot@yahoo.com](mailto:hfrumblefoot@yahoo.com) (Hambut)  
Date: 18 Jul 2001 14:54:59 -0700  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

A final point worth noting is that if everything goes horribly wrong and you have to go 'C' then I understand that it's well worth looking at:

- The MISRA 'C' Guidelines, and
- "Safer C" by Les Hatton

According to people 'in the know' both of these are supposed to have valuable tips for developing safer 'C' software.

From: [mjsilva@jps.net](mailto:mjsilva@jps.net) (Mike Silva)  
Date: 18 Jul 2001 17:30:20 -0700  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

FWIW, the MISRA 'C' Guidelines intend their C subset for SIL 2 & 3, but make no recommendation for SIL 4. I guess you could count that as a negative recommendation for C (and by implication C++)

From: [hfrumblefoot@yahoo.com](mailto:hfrumblefoot@yahoo.com) (Hambut)  
Date: 22 Jul 2001 00:04:16 -0700  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

Another reference:

Extracted from "NASA Guidebook for Safety Critical Software - Analysis and Development", NASA-GB-1740.13-96

"5.3.11.8 Programming Languages: Conclusions

A technique for evaluating safety-critical languages has been described along with other considerations for implementers and for choosing a vendor. The Ada subset we have described is suitable for safety-critical systems. Although a Pascal subset is suitable it was not described because Ada is rapidly becoming the most common language on new projects. The choice of "C" is to be avoided for our domain of interest because the language lacks the features that permit robust, reliable programming. The Ada subset minimizes the insecurities that were discussed in this entry by restricting the misuse of certain features. The objective

of the subset is to define an unambiguous syntax so that the semantics can be used for proofs if necessary. All Ada implementations have defects and implementers of safety-critical systems should review the historical defect list to verify that the problems have been addressed and fixed in a timely fashion"

## Guide for the Use of Ada in High Integrity Systems

From: "Phil Thornley"

<[phil.thornley@baesystems.com](mailto:phil.thornley@baesystems.com)>  
Date: Fri, 20 Jul 2001 07:59:31 +0100  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

> [...] there is a very useful document being produced by an ISO committee (The Annex H Rapporteur Group) called "Programming Languages - Guide for the Use of the Ada Programming Language in High Integrity Systems".

[See also "High Integrity Programming" in AUJ 20.4 (January 2000), p.252, and "ISO Guide for the Use of Ada in High Integrity Systems" in AUJ 20.2 (July 1999), p.124. -- dc]

The work on this finished some time ago and it was published early last year. The official designation is ISO/IEC DTR 15942.

The published version is under ISO/IEC copyright, so you have to buy it from your local standards source. However a very late draft version is at: <http://anubis.dkuug.dk/JTC1/SC22/WG9/n359.pdf>

I believe that this is still under ISO/IEC copyright, so I guess you can read it, but not copy it further. An earlier (but not substantially different) version that isn't under ISO/IEC copyright was published in Ada Letters - Volume XVIII Number 4, July/August 1998.

From: Peter Amey <[pna@praxis-cs.co.uk](mailto:pna@praxis-cs.co.uk)>  
Date: Fri, 20 Jul 2001 12:31:29 +0100  
Organization: Praxis Critical Systems  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

Further to the above, Jim Moore, convener of ISO WG9 has taken steps to try and get the HRG report made freely available. There is a mechanism for doing this for "technical reports" as opposed to "standards".

## Switching to Ada

From: Pascal Obry <[p.obry@wanadoo.fr](mailto:p.obry@wanadoo.fr)>  
Date: 21 Jul 2001 09:40:49 +0200  
Subject: Official recommendations of Ada  
Newsgroups: [comp.lang.ada](mailto:comp.lang.ada)

> From management's view point, using Ada instead of C/C++ is taking a major risk.

What is the major risk ?

You probably know the statistics: more than 80% of projects (at least in my domain which is Information System) are doubling the budget or the time to deliver or both.

So the risk the manager is going to take is: "Well, let me risk to have this project be delivered on time". Quite a hard to take risk, right?

I think the manager and developers don't want to switch just because they are not willing to do something different... just keep doing what we have always done.

You'll say, but the cost of learning a new language ?

I'll answer, no problem for a real and experienced developer team. Just a week of practice. The problem is not the language but the underlying principles: OO, encapsulation, abstraction, information hiding...

PS: I'm a guy who have taken the risk for a IS project :)

[And from another message: -- dc]

Just let me add that I do understand that the language is only a part of the problem. But at least switching from C++ to Ada is quite easy to do.

Other problems (far more complex to fix):

- software not well specified (over specification)
- team not trained sufficiently in general
- insufficient project management
- software not well designed
- configuration management
- too many peoples on the team
- the team does not quite understand the domain they are working on (I've seen a banking project using float for the money (under C++) because the team have been working on scientific projects before !)
- ...

## Measuring Productivity

*From: "Marin David Condic"*  
*<marin.condic@pacemicro.com>*  
*Date: Tue, 24 Jul 2001 08:34:31 -0400*  
*Subject: Software Metrics (was Official recommendations of Ada)*  
*Newsgroups: comp.lang.ada*

I would like to ask a favor of those in this thread who have an interest in productivity metrics: Please define and explain the metric(s) you would be interested in collecting. (This is not an attempt to insult or deride anyone - I'm professionally curious about what metrics have been used to measure productivity and/or what metrics might be considered valuable for measuring productivity. If you feel slammed, then submit a post to asuage your guilt! :-)

In a past life, I measured productivity in a study of control system code. (This was comparing Ada to other languages used in control system development). The numbers may not be significant for other types of systems as a result. It should be useful for other embedded/realtime apps with all/most code being developed from the ground-up. (No we didn't have an RTOS, or libraries of utilities, etc. We had (with Ada at least) the RTK that was provided with the embedded compiler and that was about it.) Without covering the development process in detail, this is basically what we did:

All software was under a CM system, so we could identify modules changed between two times. We presumed that to add a new module, delete a module or change a module (even a single line of code in it) you basically had to read and understand that module, so you got credit for the whole thing. Statistically, module size didn't vary hugely, so we didn't think we'd see much skewing of results due to modifying a single 10,000 sloc module regularly. Over time, we thought it would all average out enough to be useful.

We could easily get man-hours/month for a given project. We could relatively easily get modified modules (reduced to a semicolon count of all modules) within a month. Hence we could calculate some version of SLOCs/Fortnight. Introducing Ada and other factors moved our productivity from about 35 SLOCs/Hour to roughly 70 SLOCs/hour. (Not all credit could be given to Ada alone - there were other high level tools introduced to go with it.)

Defects were another matter. We had a change request system tied to the CM system, so it was easy to identify all change requests made in a month. The submitter of a CR and the approver of a CR both made some kind of determination of if the CR was asking for an enhancement/new feature or if it was correcting some kind of deficiency in the code. Deficiencies were chalked up as defects and counted. Introduction of Ada (and the tools) resulted in reduction of defects by a factor of four. We largely credited Ada for this (rather than the tools) because of the types of errors that were being eliminated. Most of the errors that were not being made were the kinds of things that the syntax/semantics of Ada were not allowing to occur. (Scaling errors, indexing errors, addressing errors, etc. Exactly the sort of errors that C/C++ would \*not\* catch and eliminate.)

Anyway, that outlines my experience with collecting productivity numbers and I would appreciate other ideas about how y'all would personally go about doing it. This is because I'm interested in what people would consider to be a \*meaningful\* measure of productivity. If we expect to state that Ada is superior for

productivity, we could maybe be clear about what we mean.

## U.S. Businesses Lost Nearly \$100b Due To Defective Software

*From: "Marin David Condic"*  
*<marin.condic@pacemicro.com>*  
*Date: Tue, 10 Jul 2001 09:35:25 -0400*  
*Subject: U.S. Businesses Lost Nearly \$100b Due To Defective Software*  
*Newsgroups: comp.lang.ada*

Just looking at the July issue of "Communications of the ACM" and notice a little news-blurb that claims that US businesses lost \$100 billion due to defective software code in 2000. Some of the examples they cite include eBay shutting down due to flawed Sun code and Nike losing \$100m because of supply chain management code being flawed.

Naturally none of the information is specific enough to claim that "If only everyone would use Ada..." but I think it adds a piece of evidence to cite in arguing the case for Ada. \$100 billion is starting to look like real money and if Ada could reduce error rates by - say 10% - well, you can do the math. I'd be glad to take it and retire. :-)

## 'Quick Wins' Almost Always Lose

*From: Datamation IT Management Update*  
*<listsupport@internet.com>*  
*Date: Tue, 7 Aug 2001 15:33:39 -0500*  
*Subject: Datamation, August 7: 'Quick Wins' Almost Always Lose*

[...] With the slump in the software industry showing no signs of easing, tech companies are increasingly tempted to accelerate the product production process while still tightening their belts. This can lead managers down a dangerous but common path -- cutting corners to rush a product out the door.

In this month's Quality Quest, columnist Linda G. Hayes explains why the long-term costs of a "quick win" usually will outweigh any minor short-term benefits. [...]

Chris Nerney, Executive Editor

[...]

Quality Quest: 'Quick Wins' Almost Always Lose

"Quick wins" are the technology world's equivalent of a crash diet: They produce short-term results that boomerang into a near-term eating binge and a long-term weight problem. Quality Quest columnist Linda G. Hayes, who is co-founder and CEO of automated-testing software and services provider WorkSoft, describes what happened when one software company, eager to demonstrate a "quick

win," unwisely dispensed with critical testing plans and procedures.

[http://itmanagement.earthweb.com/ecom/ecomappdev/article/0,,12170\\_861521,00.html](http://itmanagement.earthweb.com/ecom/ecomappdev/article/0,,12170_861521,00.html)

## A Frightening Story...

*From: Richard Riehle*

*<richard@adaworks.com>*

*Date: Tue, 21 Aug 2001 23:17:01 -0700*

*Organization: AdaWorks Software Engineering*

*Subject: Re: How Ada could have prevented the Red Code distributed denial of service attack.*

*Newsgroups: comp.lang.ada, comp.lang.c, comp.lang.c++, comp.lang.functional*

> Fact: there is \*NO\* Ada OS

It really depends on what you call an OS. There is certainly no OS equivalent to MS Windows, UNIX, or such. However, there are commercial, off-the-shelf (COTS) RTE's for embedded systems that serve in the role of OS for those environments. In fact, they serve in that role far better than one of the more popular OS could. The U.S. Navy is discovering just how horrid NT is after towing ships back to port because of failures. Windows XP promises to be no better.

Perhaps it is worthwhile for someone to write an OS in Ada that supports desktop applications. However, the real strength of Ada, and its real application domain is safety-critical software. The currently available COTS Operating Systems from Ada compiler publishers meets that need quite nicely, thank you.

Richard Riehle, [richard@adaworks.com](mailto:richard@adaworks.com), <http://www.adaworks.com>

*From: Markus Mottl <mottl@miss.wu-wien.ac.at>*

*Date: Wed, 22 Aug 2001 10:24:35 +0000 (UTC)*

*Organization: University of Economics and Business Administration, Vienna, Austria*

*Subject: Re: How Ada could have prevented the Red Code distributed denial of service attack.*

*Newsgroups: comp.lang.ada, comp.lang.c, comp.lang.c++, comp.lang.functional*

> Could you share a reference to a report?

Here is a somewhat longer treatment of the case: <http://www.jerrypournelle.com/reports/jerry/Yorktown.html>

In short: a divide-by-zero in a database caused a Windows NT server to crash, paralyzing the whole computer network on the cruiser Yorktown for more than two hours.

As usual, official reports (i.e. by the Navy itself) that indicate shortcomings of their weapon technology do not circulate

for too long for obvious reasons (but maybe they are just hidden well enough).

There are plenty of serious media that report on the case, e.g. Government Computing News: <http://www.gcn.com/archives/gcn/1998/december14/39.htm> [http://www.gcn.com/vol18\\_no36/com/903-1.html](http://www.gcn.com/vol18_no36/com/903-1.html)

> The "more official", the better (there are people who need convincing).

The unofficial ones are funny, too: <http://www.atlas-club.com.au/jokes/aviation/jokesav2.htm>

The frightening about some jokes is that they seem so realistic ;)

Markus Mottl, <http://miss.wu-wien.ac.at/~mottl>

## Tasking in Safety-critical Systems

*From: Roger Racine*

*<rracine@draper.com>*

*Date: Mon, 30 Jul 2001 08:19:03 -0400*

*Subject: Re: Safe Ada*

*To: "GNAT Discussion List"*

*<gnatlist@lyris.seas.gwu.edu>*

> What are the components of Ada 95 that are safe. [...] Tasking is also unsafe because of non-deterministic activity.

I do not know why you think that tasking is unsafe. It saved the Apollo 11 mission (a low-priority task was taking too long, due to a switch being in the wrong position, but the high-priority, flight-control task kept on working).

It is being used on the Space Shuttle and the International Space Station (and many other life-critical embedded systems).

[...] If you really do not want any runtime, GNAT has a pragma to create programs of that type (check the documentation for the name). There is also the Ravenscar pragma for the subset created by the International Workshop on Real-time Ada.

*From: Roger Racine*

*<rracine@draper.com>*

*Date: Tue, 31 Jul 2001 07:33:37 -0400*

*Subject: Re: Safe Ada*

*To: "GNAT Discussion List"*

*<gnatlist@lyris.seas.gwu.edu>*

> It is quite complicated to prove that even simple programs involving tasking have some desirable property (and no undesirable ones).

It does, of course, depend on the application, but I disagree completely for inherently parallel, cyclic programs such as guidance, navigation and control systems. Especially with Ada, where there is much less likelihood of memory corruption, tasking systems are every bit as easy to verify as non-tasking systems. And they are much, much easier to develop, integrate, and maintain.

Non-tasking programs that have cyclic activities at different rates need to have the low-rate activities broken up into pieces that can fit in the highest rate period. This is generally a trial and error process, and is a nightmare to change.

For a tasking system, all you need to do is make sure all your deadlines are met (there is a nice formula, based on the execution times of the tasks), and be careful about sharing data. Maintenance is much simpler.

There has been a good deal of research on this topic. And, as I said in my previous message, tasking has been used successfully on many complex, safety-critical systems (Apollo, Space Shuttle, and International Space Station, to name three). It is difficult to break down antique rules [...], as it was in the commercial aviation world in the U.S. (the FAA used to have a similar rule [prohibiting tasking -- dc], but they have been convinced).

## Does Ada Need Garbage Collection?

*From: Mark Lundquist*

*<mlundquist2@home.com>*

*Date: Mon, 20 Aug 2001 11:50:18 -0400*

*Subject: Re: Does GNAT Ada 95 provide garbage collector ?*

*To: team-ada@acm.org*

> If Ada does not have a garbage collection, does it need one?

From a marketing perspective, the answer may be "yes".

In the software industry overall, garbage collection is high on the "must have" list for an implementation language. Why? Because memory leaks are such a huge problem. People can argue forever about \*why\* memory leaks are in fact such a big problem, but in the eyes of many, managing the lifetimes of dynamically allocated objects is just one more burden from which programmers should be relieved. Are they right about this? Who cares, because right now we are talking about the marketing perspective :-)

Open Source advocate Eric Raymond discusses languages in a chapter of his book in progress, "The Art of Unix Programming". The chapter is entitled "To C or not to C?" and you can read it at <http://www.tuxedo.org/~esr/writings/taoup/chapter3.html>. Notice the primacy given to automatic storage management as the distinguishing feature of modern languages, or languages that have a future. It practically trumps all other considerations. While I don't share this perspective, I would guess that it represents the views of many.

So, if there are prospective users for whom GC is the make-or-break issue, the availability of a GC Ada implementation might help them to at least consider Ada.

But there are some problems with GC as well. One problem is that you must pay something for it in system overhead. You also may have to sacrifice schedulability as well, you're willing to pay in overhead: real-time (deterministic) GC can be done, but it is slower.

In the culture of IT geekdom, there is really no point in asking whether the penalties for GC are worth it, since it seems to really be the only cure for the plague of memory leaks (but not the only treatment... in recognition of the practice of periodically restarting application servers to preempt the inevitable memory-leak-induced crash, the ASP.NET platform provides for -- automatic scheduled restarts! LOL...). If you ask, "but where will the extra cycles come from?" the answer is, "From Intel." That's what we have Intel for, to crank out ever-faster machines to accommodate things like GC, right?

Someone always objects with some statement like, "Oh yeah, but what about if you have, like, an airplane, and the collision-avoidance alarm is about to go off but then the system has to stop for the GC to run, and the planes crash into one another and everybody perishes in a giant fireball... "where's my Mommy, I want my Mommy!" "Sorry, son, that's the price you pay for GARBAGE COLLECTION... at least it wasn't a MEMORY LEAK..."

Except we're not talking about Ada in airplanes here. It's already in airplanes. We're talking about getting Ada into the stuff people buy on Egghead.co -- oops. OK, the stuff people buy from Amazon.com and Office Depot. You know, the other 95% of software that doesn't care if the GC has to run every so often.

The other problem is that there are other types of resource leaks besides memory leaks. Anything that has to be claimed and then released is a potential source of

leaks: synchronization objects like mutexes, file descriptors, connections to session-based services, etc.

GC doesn't cure these problems; moreover, other problems arise around the attempts to link finalization/destruction with garbage collection, which always falls short and must be shored up with hacks, and even with the hacks you end up with something that is not altogether satisfactory (e.g. Java, .NET).

> or does it just manage it's objects by their scopes and global availability?

Something like that. Ada gives you kind of a cocktail of features that work together to help guard against memory leaks:

1) When you declare an access type, a storage pool is created for that access type. When the access type goes out of scope, the storage pool is freed. This ensures that a dynamically allocated object can't live longer than the access type that was used to allocated it.

2) Accessibility checks (such as for access parameters). These close some loopholes that can otherwise result in dangling pointers.

3) Controlled types, which allow you to do things like reference-counted storage management, by specifying whatever needs to be done for initialization, assignment and finalization.

> By that I am asking is would it be an actual advantage to have a garbage collector in the language?

I would say, it would have just about as much actual "silver bullet factor" as in any other language, which in turn is less than it's made out to be.

## A Few Quotes

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

*Date: Thu, 23 Aug 2001 11:12:53 +0200*

*Organization: Adalog*

*Subject: Re: Subtle Bugs, kudos Ada (was How Ada ...Red Code ...)*

*Newsgroups: comp.lang.ada, comp.lang.c, comp.lang.c++*

> Would you hire the programmer who wrote this function, to work with you on a project in \*any\* programming language?

Yes, programmers make errors.

Two quotes that I love to bring together:

From one of the first books about C by K&R:

"C was designed on the assumption that the programmer is someone sensible who knows what he's doing"

From the introduction of the Ada Reference Manual:

"Ada was designed with the concern of programming as a human activity"

The fact that these starting hypothesis lead to two completely different philosophies of languages is left as a subject for meditation...

*From: Gary Scott*

*<Gary.L.Scott@lmtas.lmco.com>*

*Date: Thu, 23 Aug 2001 19:29:01 -0500*

*Organization: LM Aeronautics*

*Subject: Commentary on Industry Move to C++*

*Newsgroups: comp.lang.ada*

I've had some time now to consider the recent trends of my industry moving new projects to C++. As a lead system test engineer, I am now inclined to approve of this trend. I anticipate much more business (testing) in the future due to a significant increase in the number of software anomalies. This I feel will keep me gainfully employed for a number of additional years while also making my job much more interesting (more challenging anomalies to solve).

# Conference Calendar

This is a list of European and large world-wide events that may be of interest to the Ada community. More information on items marked ♦ is available elsewhere in the *Journal*. The information here is extracted from the online *Conference announcements for the international Ada community* at <http://www.cs.kuleuven.ac.be/~dirk/ada-belgium/events/list.html> on the Ada-Belgium webserver. These pages contain full announcements, calls for papers, calls for participation, programmes, URLs etc and are updated regularly.

---

## 2001

- 30 Sept. – 04 October     **2001 ACM SIGAda Annual International Conference (SIGAda'2001)** Twin Cities, Minnesota, USA. Sponsored by ACM's special interest group on the Ada Programming Language (SIGAda), in cooperation with Ada-Europe and ACM SIGAPP, SIGCAS, SIGCSE, SIGPLAN, SIGSOFT, and Twin Cities ACM SIGAda.
- 01-05 October            **4th International Conference on the Unified Modeling Language (UML'2001)** Toronto, Canada Theme: "Modeling Languages, Concepts and Tools"
- 02-05 October            **8th Working Conference on Reverse Engineering (WCRE'2001)** Stuttgart, Germany Topics include: Experience reports (successes and failures) on reverse engineering or reengineering efforts; Techniques, tools, and enabling technologies for reengineering, reverse engineering, renovation, reuse, and migration; Software visualization; Software evolution and reengineering; Integration of reverse engineering and forward engineering; Code-based management systems to support reverse engineering; Wrapping and interfacing legacy systems; Formal methods in reverse engineering; etc.
- 03-05 October            **15th Brazilian Symposium on Software Engineering (SBES'2001)** Rio de Janeiro, Brazil. Topics include: Industrial applications of Software Engineering; Component-based Software Engineering; Methods, Techniques, Languages and Tools for Software Engineering; Software Maintenance; Software Quality; Software verification, validation and testing; etc.
- 08-12 October            **25th Anniversary Annual International Computer Software and Applications Conference (COMPSAC'2001)** Chicago, Illinois, USA Theme: Invigorating Software Development Topics include: Component-based software development; Object-oriented technology; Safety and security; Software reliability; Distributed systems; Embedded systems; Internet and Web-based systems; Middleware systems; etc.
- 08-12 October            **International Conference on Practical Software Quality Techniques (PSQT'2001 North)** St Paul, Minnesota, USA. Theme: "More Reliable Systems for the Internet"
- 09-11 October            **Embedded Systems Conference Europe (ESC Europe'2001)** Stuttgart, Germany.
- 10-12 October            **European Software Process Improvement Conference (EuroSPI'2001)** Limerick Institute of Technology, Ireland.
- 14-18 October            **16th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'2001)** Tampa Bay, Florida, USA Deadline for submissions: July 20, 2001 (Posters, Demonstrations, Doctoral Symposium, and Student Volunteers)
- 15-19 October            **Absolute Software – Public Ada Course** Carlsbad, California, USA. Topic: Ada 83 (Ada 95 on 2001/08/20-24)
- 17-19 October            **Colloque Francophone sur la Modelisation des Systemes Reactifs (MSR '2001)** Toulouse, France
- ♦ 24 October             **Symposium on Reliable Object-Oriented Programming (SROOP)** London, UK. OO has become a key feature of system design and implementation. Reliable systems are increasingly using OO techniques, often replacing traditional structured approaches. Programming languages like Ada 95, Java and C++ each offer subtly different ways of representing objects, their attributes and their methods. How can those features be used reliably? Are there features that are unsafe or

inappropriate? What architectures can be employed to make implementation easier or more verifiable? Where do patterns and frameworks fit in?. This Symposium will address many of these issues and more.

- 28-31 October **20th IEEE Symposium on Reliable Distributed Systems (SRDS'20)** New Orleans, USA Topics include: Distributed systems with reliability, availability, security, safety, and/or real-time requirements; Security and High Confidence Systems; Formal methods and foundations for reliable distributed computing; Distributed objects and middleware systems; Distributed and Web-based application systems; etc.
- 31 Oct. – 2 Nov. **8th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'2001)** Bologna, Italy. Topics include: Parallel/Distributed Systems; Security, Safety and Assurance of Distributed Systems; Applications of Distributed Systems; Distributed Object Computing; etc.
- 06-10 November **IEEE International Conference on Software Maintenance (ICSM'2001)** Florence, Italy Theme: "Systems and Software Evolution in the era of the Internet" Topics include: Design for maintenance; Internet and distributed systems; Software reusability; Tools and environments; Commercial off-the-shelf (COTS); Freeware and open source applications; Programming languages; etc.
- 12-16 November **John Robinson & Associates - Public Ada Training Course** Bradford, UK. A five day training course with streams for Ada 83 and Ada 95.
- 12-16 November **5th International Internet & Software Quality Week Europe** Brussels, Belgium. Theme: "Internet NOW!" Topics include: Productivity and Quality Issues; Process Improvement; Real-Time Software; Object Oriented Testing; Application of Formal Methods; Cost/Schedule Estimation; Software Reliability Studies; E-Commerce Reliability; Quality of Service (QoS); Risk Management; etc.
- 12-16 November **IFIP/ACM International Conference on Distributed Systems Platforms (Middleware'2001)** Heidelberg, Germany Topics include: integration of middleware platforms with web and Java technologies; real-time middleware platforms including real-time ORBs; reliable middleware platforms including fault-tolerant ORBs; applications of middleware technologies including telematics and commerce; distributed systems management and interactive configuration and development tools; etc
- 16-17 November **4th International Conference on Compilers, Architectures and Synthesis for Embedded Systems (CASES'2001)** Atlanta, GA, USA Topics include: New optimizing compilers for embedded-domain constraints; Synergy between extant parallel computing technologies, such as notations for expressing concurrency, and instruction level parallel processing; etc.
- 22-23 November **6th Australian Workshop on Requirements Engineering (AWRE'2001)** Sydney, Australia.
- 26-29 November **16th IEEE International Conference on Automated Software Engineering (ASE '2001)** San Diego, USA.
- 27-30 November **12th International Symp. on Software Reliability Engineering (ISSRE'2001)** Hong Kong.
- 30 November **1st International Workshop on Model-based Requirements Engineering (MBRE'01)** San Diego, USA. Co-located with the 16th IEEE International Conference on Automated Software Engineering (ASE'2001).
- 03-06 December **22nd IEEE Real-Time Systems Symposium (RTSS'01)** London, UK. Topics include: embedded systems, software engineering, programming languages and run-time systems, middleware systems, design and analysis tools, formal methods, case studies, applications, etc. Including:
- 03 December **Workshop on Real-Time Embedded Systems**
- 04-06 December **14th International Conference on Software & Systems Engineering and their Applications (ICSSEA'2001)** Paris, France.
- 04-07 December **8th Asia-Pacific Software Engineering Conference (APSEC'2001)** Macau, China. Topics include: component-based design techniques, concurrent systems, design patterns, distributed systems, formal methods, object-oriented analysis and design, programming languages, reactive and real-time embedded systems, reengineering and reverse engineering, software. quality,



software reusability, software maintenance, software engineering for the Internet and the E-commerce, software engineering education, tools and environments, etc.

- 10 December **Birthday of Lady Ada Lovelace, born in 1815 – Happy Programmers' Day!**
- 10-11 December **2nd Asia-Pacific Conference on Quality Software (APAQS'2001)** Hong Kong. Topics include: Economics of software quality and testing; Performance and robustness testing; Quality evaluation of software products and components; Reliability; Review, inspection, and walkthrough; Software quality education; Static and dynamic analysis; Testing of object-oriented systems and real-time systems; Validation and verification; Application areas such as component-based systems, distributed systems, embedded systems, information systems, etc.
- 17-20 December **8th International Conference on High Performance Computing (HiPC'2001)** Hyderabad, India. Topics include: Parallel Languages & Compilers; Distributed Systems; Programming Environments; Embedded Systems; etc.

---

## 2002

- 16-18 January **29th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'2002)** Portland, Oregon, USA.
- 19 January **9th International Workshop on Foundations of Object-Oriented Languages (FOOL 9)** Portland, Oregon, USA. Topics include: language semantics, type systems, program analysis and verification, concurrent and distributed languages, etc. Deadline for submissions: October 12, 2001.
- 23-25 January **8th International Conference on Languages and Models with Objects (LMO'2002)** Montpellier, France. Topics include (in French): Programmation par objets (Langages, interpretation, compilation; modeles d'objets pour la programmation; objets et types; environnements de programmation; etc.); Composants et objets en reseau (Modeles de composants a objets; interactions de composants; developpement a base de composants, composants reutilisables; objets et composants distribues, repartis; acteurs, parallelisme; objets et internet; interoperabilite); Genie des objets (Cycle de vie des objets; retro-conception, evolution des programmes, versions; surete des programmes, specifications formelles; methodes d'analyse et de conception objet, UML; ingenierie des modeles et des meta-modeles; reutilisation, architectures logicielles reutilisables et a base de composants; hierarchies, frameworks, patterns); Applications; etc.
- 18-21 February **14th Software Engineering Process Group Conference (SEPG'2002)** Phoenix, Arizona, USA.
- 25-27 February **15th Conference on Software Engineering Education and Training (CSEET'2002)** Covington, Kentucky (Greater Cincinnati), USA.
- 04-08 March **International Conference on Practical Software Quality Techniques & Testing Techniques (PSQT/PSTT'2002 South)** New Orleans, USA. Deadline for submissions: October 14, 2001 (tutorials), November 4, 2001 (papers).
- 06-08 March **Ada-Deutschland Tagung 2002** Jena, Germany. Deadline for submissions: November 23, 2001.
- 10-13 March **2002 ACM Symposium on Applied Computing (SAC'02)** Madrid, Spain. Deadline for submissions:
- 11-13 March **5th International Conference on "Achieving Quality In Software" (AQUIS'2002)** Venezia, Italy.
- 14-15 March **2nd International Software Process Improvement and Capability determination Conference (SPICE'2002)** Venice, Italy.
- 20-22 March **5th IFIP International Conference on Formal Methods for Open Object-based Distributed Systems (FMOODS'2002)** Twente, The Netherlands. Topics include: Specification and analysis techniques for distributed systems; Semantics of object-based programming languages; Design and software life-cycle of object-based distributed applications; Applications to telecommunications and related areas; etc.

- 06-14 April **European Joint Conferences on Theory and Practice of Software (ETAPS'2002)** Grenoble, France. Includes:
- 06-14 April **8th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2002)** Topics include: Verification and construction techniques; Compositional and refinement-based methodologies; Analytical techniques for real-time, hybrid and safety-critical systems; Tool environments and tool architectures; Applications and case studies; etc. Deadline for paper submissions: October 19, 2001.
  - 07 April **International Conference on Compiler Construction (CC'2002)** Topics: compiler construction, programming language implementation and language design Deadline for paper submissions: October 19, 2001 .
  - 13 April **2nd Workshop on Language Descriptions, Tools and Applications (LDTA'2002)** Deadline for paper submissions: December 15, 2001
  - 13 April **Synchronous Languages, Applications, and Programming (SLAP'2002)** Deadline for paper submissions: November 13, 2001.
- 07 April **Workshop on Software Composition (SC'2002)** Deadline for paper submissions 1 October 2001.
- 07-10 April **10th Object Technology Conference (OT'2002)** Oxford UK. Topics include: Component technology, Languages, Distributed systems, Small and embedded systems, Patterns, Lessons learned/experience reports etc.
- 08-11 April **9th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS'2002)** Lund, Sweden. Topics include: Component-based Design and Reuse; Applied Formal Methods and Security; Tools and Environments; Education and Training; Embedded Systems; Reliability, Dependability, Safety; Verification and Validation; Standards; etc. Deadline for submissions: October 15, 2001 (papers and workshop proposal).
- ◆ 09-12 April **11th International Real-Time Ada Workshop (IRTAW'2002)** Mont-Tremblant, Quebec, Canada. Deadline for position paper submissions: November 1, 2001.
- 15-19 April **7th International Conference on Software Reuse (ICSR-7)** Austin, Texas, USA. Topics include: Software product lines and product line architectures; Component-based software engineering; Lightweight approaches to software reuse; Quality aspects of reuse, e.g. security and reliability; Success and failure stories of reuse approaches from industrial context; etc. Deadline for submissions: October 9, 2001 (papers and panel proposals), November 12, 2001 (tutorials), January 30, 2002 (posters and demos).
- 15-19 April **International Parallel and Distributed Processing Symposium (IPDPS'2002)** Fort Lauderdale, Florida, USA. Topics include: Applications of parallel and distributed computing, including web applications and scientific applications; Parallel and distributed software, including parallel programming languages and compilers, operating systems, schedulers, runtime, middleware, libraries, programming environments and tools for parallel and distributed computing; etc. Deadline for submissions: September 21, 2001 (papers), October 20, 2001 (tutorials, industrial track). Includes:
- 15-19 April **7th International Workshop on Formal Methods for Parallel Programming: Theory and Applications (FMPPTA'2002)** Deadline for submissions: October 29, 2001
- 17-19 April **3rd International Conference on Software Testing (ICSTEST'2002)** Dusseldorf, Germany.
- 21-24 April **2nd International Conference on Computational Science (ICCS'2002)** Amsterdam, The Netherlands. Topics include: Parallel and Distributed Computing; Problem Solving Environments (including: Software Component Technology); Education in Computational Science; etc. Deadline for submissions: November 1, 2001.
- 23-26 April **1st International Conference on Aspect-Oriented Software Development (AOSD'2002)** Enschede, The Netherlands. Topics include: language design and implementation; analysis, design and development tools; software engineering; lifecycle support; etc. Deadline for submissions:

- October 8, 2001 (papers), October 30, 2001 (workshops, tutorials), January 15, 2002 (demonstrations).
- 29 April – 01 May **5th IEEE International Symposium on Object-oriented Real-time distributed Computing (ISORC'2002)** Washington DC, USA.
- 19-25 May **International Conference on Software Engineering (ICSE'2002)** Buenos Aires, Argentina  
Deadline for submissions: October 21, 2001 (tutorials, workshops), December 15, 2001 (posters, demos).
- 04-07 June **8th International Symposium on Software Metrics (Metrics'2002)** Ottawa, Canada Theme: "Measuring and Managing Software Risks in the Age of Internet"
- 09-12 June **7th European Conference on Software Quality** Helsinki, Finland. Deadline for submissions: November 15, 2001 (papers), December 31, 2001 (workshops, panels, tutorials, posters, projects)
- 09-14 June **27th Annual USENIX Technical Conference (USENIX'2002)** Monterey, Canada Topics include: Reliability and QoS; Usage studies; Web technologies; Interoperability of heterogeneous systems; special track on freely redistributable technology (GNOME, GNU, Linux, Tcl/Tk and more); etc. Deadline for submissions: November 12, 2001 (Freenix track), November 19, 2001 (other sessions)
- 17-19 Jun **ACM SIGPLAN 2002 Conference on Programming Language Design and Implementation (PLDI'2002)** Berlin, Germany Sponsored by ACM SIGPLan in cooperation with ACM SIGSoft Topics include: implementations of language features; language support for security and safety; techniques for embedded and mobile code; compilation for distributed, heterogeneous systems; languages and compilers for parallel computing; etc. Deadline for paper submissions: November 16, 2001
- 17-21 June **8th International Symposium on Software Metrics (Metrics'2002)** Ottawa, Canada. Theme: "Measuring and Managing Software Risks in the Age of Internet"
- ◆ 17-21 June **7th International Conference on Reliable Software Technologies - Ada-Europe'2002** Vienna, Austria. Sponsored by Ada-Europe, in cooperation with ACM SIGAda Deadline for paper, tutorial, workshop, poster submissions: October 31, 2001. 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 embedded systems, including the use of Ada in this realm.
- 23-26 June **International Conference on Dependable Systems and Networks (DSN'2002)** Washington, D.C., USA. Deadline for submissions: November 19, 2001 (papers and workshops), January 14, 2002 (tutorials)
- 20-23 August **13th International Conference on Concurrency Theory (CONCUR'2002)** Brno, Czech Republic Topics include: concurrency related aspects of: real-time systems, distributed programming, object-oriented programming, case studies, tools and environments for programming and verification, etc. Deadline for submissions: December 1, 2001 (workshops), March 25, 2002 (papers)
- 26-28 August **International Conference on Pervasive Computing (PERVASIVE'2002)** Zurich, Switzerland. Deadline for submissions: February 22, 2002 (papers and demos), June 19, 2002 (posters and short papers)
- 10 December **Birthday of Lady Ada Lovelace, born in 1815 – Happy Programmers' Day!**

# SYMPOSIUM ON RELIABLE OBJECT ORIENTED PROGRAMMING

Institution of Electrical Engineers, London WC2 : Wednesday 24 October 2001

*Object Oriented Programming, once an over-hyped niche activity, is now widely used across a broad spectrum of the software industry. The major challenge today in OOP applications is to achieve long-term reliability. In theory, the main characteristics of OO: abstraction, encapsulation, modularity and hierarchy, should support this aim. In practice, OO programs tend to be written and constructed in a very different way from programs created using classical structured approaches, and make the job of both achieving and proving reliability significantly harder. This is the case even with Ada 95, an OO language developed specifically for reliable systems.*

*This symposium addresses advances that have been made in the practical application of OO techniques and in reliable architectures based on OO principles. It will appeal equally to software engineers and to software managers who are involved in developing reliable systems.*

## Programme

09.00 Registration  
10.00 Symposium opens

### Introduction to OO Features of Ada 95

John Robinson, *John Robinson & Associates*

### Requirements for Reliable Architectures

Brian Tooby & Stuart Curtis

BAE SYSTEMS Avionics Ltd

### Object Oriented Ravenscar Profile

Brian Dobbing, *Praxis Critical Systems*

### Improving the Reliability of Object-Oriented Architecture Designs

Robert G. Pettit IV

*The Aerospace Corporation, USA*

### GENESYS An application of OO Technology to Aircraft Display Systems

Neil Davidson, *BAE SYSTEMS Avionics Ltd*

### OO Software Testing – some problems and solutions

Ian Gilchrist, *IPL*

### TUTORIAL : Idioms for Constructing Reliable OO Systems in Ada 95

Matthew Heaney

*On2 Technologies, Inc. USA*

This tutorial presents idioms for reliable class-wide programming in Ada 95, including techniques for memory-management such as factory functions, reference-counted pointers, and custom storage pools.

### Mapping UML to Ada

Bill Taylor, *Bill Taylor Associates*

### High Integrity Java – Oxymoron or Reality?

Brian Dobbing, *Praxis Critical Systems*

17.00 Symposium ends

## Application

Name/s \_\_\_\_\_ Affiliation \_\_\_\_\_

Address \_\_\_\_\_

Tel \_\_\_\_\_ Fax \_\_\_\_\_

E-Mail \_\_\_\_\_ Dietary requirements \_\_\_\_\_

**Fees:** Member rate £40 + VAT = £47 Non-Member rate £80 + VAT = £94

Ada UK regrets that cancellations will be subject to the full fees. Substitutions will be accepted.

**Booking:** number of places \_\_\_\_\_

**Payment** (please tick appropriate box/es)

Invoice my Company, using PO number \_\_\_\_\_

MasterCard or Visa payment.

Card number \_\_\_\_\_ Expiry \_\_\_\_\_ Name on card \_\_\_\_\_

**Please copy this page, then complete and return by post or fax to**

Ada Language UK Ltd, PO Box 322, York YO10 3GY, fax: +44 (0)1904 426702.

For enquiries about the Symposium, please telephone +44 (0)1904 412740 or e-mail admin@adauk.org.uk.

## ***Call for Participation***

# **11th International Real-Time Applications Workshop IRTAW 11**

**9 -12 April 2002, Mont-Tremblant, Quebec, Canada**

Over the last decade and 1/2, the International Real-Time Ada Workshops have provided a focus for identifying issues with Ada 83 and 95, proposing solutions for those problems and evaluating proposed language changes.

Since the standardization of Ada95, the International Real Time Ada workshops have assisted in the review of the real time portions of the Guidance in the Use of the Ada Programming Language for High Integrity Systems, and has developed the Ravenscar Tasking Profile.

With the advent of Java and the development of Real Time specifications for Java, the workshop has begun to consider the integration of embedded Ada and Java systems, and their interoperability.

The goals of the 11th IRTAW are to:

- examine and develop paradigms for using Ada 95 for real-time single processor, multiprocessor and distributed systems (including issues of hard and flexible scheduling);
- consider reports on experiences with using Ada 95 on actual real-time projects;
- identify the benefits and impacts of using object-oriented programming in multi tasking (potentially distributed) real-time systems;
- explore the use of Ada 95 in developing multi-tasking components which are resilient to software design errors and hardware failures;
- refine criteria for the use of Ada 95 in high integrity systems, especially those with real-time or embedded attributes;
- review the interactions between exceptions in Ada with those from other languages such as Java and C++;
- examine the issues around the interoperability of Ada and real time Java embedded systems.

Participation at the Workshop is by invitation following the submission of a Position Paper addressing one or more of the above topics by 1 November 2001. Position papers should be between five and ten pages. All accepted papers will appear in the Workshop Proceedings which will be published as a special edition of Ada Letters.

### **How to submit:**

Please submit your Position Paper to the Program Chair, Joyce Tokar, <tokar@attglobal.net> preferably in Word Perfect or Word format. Or mail four paper copies to

Dr. Joyce L. Tokar  
PO Box 1352  
Phoenix AZ 85001-1352  
USA

All the papers must be written in English. The top of the first page of the paper should include the title of the paper, the author(s)'s name(s), position, organization, address, telephone number(s), fax number(s), and the email address for the author responsible for correspondence.

### **Schedule:**

Receipt of Position Paper	1 November 2001
Notification of Acceptance	15 December 2001
Final Copy of Position Paper	1 February 2002
Workshop Date	9-12 April 2002



# CALL FOR PAPERS

## 7<sup>th</sup> International Conference on Reliable Software Technologies - Ada-Europe 2002

17 - 21 June 2002, Vienna, Austria

<http://www.ada-europe.org/conference2002.html>

### Conference Chair

*Gerhard H. Schildt*  
Technical University Vienna  
Dept. of Computer-Aided Automation  
Treitlstr. 1-3  
A-1040 Vienna, Austria  
[Schildt@auto.tuwien.ac.at](mailto:Schildt@auto.tuwien.ac.at)

### Program Co-Chairs

*Johann Blieberger*  
Technical University Vienna  
Dept. of Computer-Aided Automation  
[Blieberger@auto.tuwien.ac.at](mailto:Blieberger@auto.tuwien.ac.at)

*Alfred Strohmeier*  
Swiss Fed. Inst. of Technology Lausanne  
Software Engineering Lab  
CH-1015 Lausanne EPFL, Switzerland  
[Alfred.Strohmeier@epfl.ch](mailto:Alfred.Strohmeier@epfl.ch)

### Tutorial Chair

*Helge Hagenauer*  
University of Salzburg  
Dept. Comp. Science & System Analysis  
[hagenau@cosy.sbg.ac.at](mailto:hagenau@cosy.sbg.ac.at)

### Exhibition Chair

*Thomas Gruber*  
Austrian Research Centers Seibersdorf  
Phone +43-5-0550-4106  
Fax +43-5-0550-4199  
[thomas.gruber@arcs.ac.at](mailto:thomas.gruber@arcs.ac.at)

### Publicity Chair

*Dirk Craeynest*  
OFFIS nv/sa & K.U.Leuven  
Weiveldlaan 41/32  
B-1930 Zaventem, Belgium  
[Dirk.Craeynest@cs.kuleuven.ac.be](mailto:Dirk.Craeynest@cs.kuleuven.ac.be)

### Local Organization Chair

*Bernd Burgstaller*  
Technical University Vienna  
Dept. of Computer-Aided Automation  
Treitlstr. 1-3  
A-1040 Vienna, Austria  
[Burgstaller@auto.tuwien.ac.at](mailto:Burgstaller@auto.tuwien.ac.at)

In cooperation  
with



SIGAda



TECHNISCHE UNIVERSITÄT WIEN

### General Information

The 7th International Conference on Reliable Software Technologies (Ada-Europe 2002) will take place in the year 2002 in Vienna, Austria. The full conference will comprise a three-day technical program and exhibition from Tuesday to Thursday, and parallel workshops and tutorials on Monday and Friday.

### Schedule

31 October 2001	Submission of papers, extended abstracts and proposals for tutorials and workshops
10 January 2002	Notification to authors
10 February 2002	Full papers required for accepted extended abstracts
10 March 2002	Final papers (camera-ready) required
17-21 June 2002	Conference

### Topics

The conference will provide an international forum for researchers, developers and users of reliable software technologies. Presentations and discussions will cover applied and theoretical work currently conducted to support the development and maintenance of software systems. Participants will include practitioners and researchers from industry, academia and government. There will be a special session on embedded systems, including the use of Ada in this realm.

For papers, tutorials, and workshop proposals, the topics of interest include, but are not limited to:

- **Embedded Systems** (special session).
- **Management of Software Development and Maintenance:** Methods, Techniques and Tools.
- **Software Quality:** Quality Management and Assurance, Risk Analysis, Program Analysis, Verification, Validation, Testing of Software Systems.
- **Software Development Methods and Techniques:** Requirements Engineering, Object-Oriented Technologies, Formal Methods, Software Management Issues, Re-engineering and Reverse Engineering, Reuse.
- **Software Architectures:** Patterns for Software Design and Composition, Frameworks, Architecture-Centered Development, Component and Class Libraries, Component Design.
- **Tools:** CASE Tools, Software Development Environments, Compilers, Browsers, Debuggers.
- **Kinds of Systems:** Real-Time Systems, Distributed Systems, Fault-Tolerant Systems, Information Systems, Safety-Critical and Secure Systems.
- **Applications** in Multimedia and Communications, Manufacturing, Robotics, Avionics, Space, Health Care, Transportation, Industry.
- **Ada Language and Tools:** Programming Techniques, Object-Oriented Programming, New Approaches in Tool Support, Bindings and Libraries, Evaluation and Comparison of Languages, Language Extension Proposals.
- **Ada Experience Reports:** Experience Reports from Projects using Ada, Management Approaches, Metrics, Comparisons with past or parallel Experiences in non-Ada Projects.
- **Education and Training.**
- **Case Studies and Experiments.**

## Program Committee

Ángel Álvarez, Technical University of Madrid, Spain  
Lars Asplund, Uppsala University, Sweden  
Neil Audsley, University of York, UK  
Stéphane Barbey, Paranor AG, Switzerland  
John Barnes, UK  
Guillem Bernat, University of York, UK  
Maarten Boasson, University of Amsterdam, the Netherlands  
Ben Brosgol, Ada Core Technologies, USA  
Bernd Burgstaller, Technical University Vienna, Austria  
Ulf Cederling, Växjö University, Sweden  
Roderick Chapman, Praxis Critical Systems Limited, UK  
Paolo Coppola, INTECS HRT, Italy  
Dirk Craeynest, OFFIS nv/sa & K.U.Leuven, Belgium  
Alfons Crespo, Universidad Politécnica de Valencia, Spain  
Peter Dencker, Aonix GmbH, Germany  
Raymond Devillers, Université Libre de Bruxelles, Belgium  
Jesús M. González-Barahona, Universidad Rey Juan Carlos, Spain  
Michael González Harbour, Universidad de Cantabria, Spain  
Helge Hagenauer, University of Salzburg, Austria  
Günter Hommel, Technische Universität Berlin, Germany  
Jan van Katwijk, Delft University of Technology, the Netherlands  
Hubert B. Keller, Forschungszentrum Karlsruhe, Germany  
Yvon Kermarrec, ENST Bretagne, France  
Jörg Kienzle, Swiss Fed. Inst. of Technology Lausanne, Switzerland  
Fabrice Kordon, Université P. & M. Curie, France  
Albert Llamós, Universitat de les Illes Balears, Spain  
Kristina Lundqvist, MIT, USA  
Franco Mazzanti, Ist. di Elaborazione della Informazione, Italy  
John W. McCormick, University of Northern Iowa, USA  
Pierre Morere, Aonix, France  
Laurent Pautet, ENST Paris University, France  
Erhard Plödereder, University of Stuttgart, Germany  
Juan A. de la Puente, Universidad Politécnica de Madrid, Spain  
Gerhard Rabe, TÜV Nord e.V., Hamburg, Germany  
Jean-Marie Rigaud, Université Paul Sabatier, France  
Alexander Romanovsky, University of Newcastle, UK  
Jean-Pierre Rosen, Adalog, France  
Bo I. Sandén, Colorado Technical University, USA  
Gerhard H. Schildt, Technical University Vienna, Austria  
Bernhard Scholz, Technical University Vienna, Austria  
Edmond Schonberg, New York University & ACT, USA  
Tullio Vardanega, European Space Research and Technology Center, the Netherlands  
Stef Van Vlierberghe, Eurocontrol CFMU, Belgium  
Andy Wellings, University of York, UK  
Ian Wild, Eurocontrol CFMU, Belgium  
Jürgen Winkler, Friedrich-Schiller-Universität, Germany  
Thomas Wolf, Paranor AG, Switzerland

11 Sept. 2001, BLIEB/AS, 09/A4

## Submissions

Authors are invited to submit original contributions. Submissions should be in English. An extended abstract (4-6 pages) or, preferably, the full paper (up to 12 pages) should be sent using the Web submission form. For more information please see the conference Web page. Submissions should be in PDF, Postscript or ASCII format, and follow the LNCS instructions (see 'Proceedings' below). Submissions by other electronic formats, such as a word processor source file, or by fax are not accepted. The Web submission form is the preferred procedure. However, if you don't have access to the Internet, or you don't have an appropriate Web browser, you may send your extended abstract or paper by e-mail to the Program Co-Chair Johann Bliederger. If electronic submission is not available, please send five paper copies. For e-mail or paper submissions, the body of the electronic message or the first page should identify the submission as a paper or extended abstract, and should include in plain text: the title; name, current affiliation, postal address, e-mail address, telephone and fax of each author; the name of the designated contact person; a short abstract; and a list of keywords ordered by relevance, including, whenever possible, topics and subtopics taken from the list of topics of the conference. If the paper is submitted in Postscript, please be sure to select the option "optimize for portability" in your printer driver. To enable publication of the accepted papers before the conference, strict adherence to the dates of the schedule is essential. Authors of extended abstracts must be prepared to submit a full version of their paper before February 10, 2002.

## Proceedings

The proceedings will be published in the Lecture Notes in Computer Science (LNCS) series by Springer Verlag, and will be available at the start of the conference. See the LNCS Authors Instructions page for guidelines (<http://www.springer.de/comp/lncs/authors.html>).

## Awards

There will be honorary awards for the best paper and the best presentation.

## Call for Tutorials

A tutorial should address any of the topics of the theme of the conference. A tutorial will last a half or full day. The proposals should include a title, an abstract, a description of the topic, a detailed outline of the presentation, a description of the presenter's teaching experience in general and with the proposed topic, duration (half day or full day), level of the tutorial (introductory, intermediate, or advanced), expected audience experience and background. Proposals should be submitted by e-mail to the Tutorial Chair.

## Call for Workshops

Half- and full-day workshops can be held to address timely issues or to initiate a longer term effort on a topic of interest. Proposals should be submitted by e-mail to the Program Co-Chair Alfred Strohmeier.

## Exhibition

The conference will be accompanied by a three-day commercial exhibition on June 18, 19 and 20. Vendors of software products and services should contact the Exhibition Chair at their earliest convenience for further information and to ensure their inclusion.

# Guide for the use of the Ada Ravenscar Profile in high integrity systems – the work of the HRG

*Alan Burns*

*Chair HRG, Department of Computer Science, University of York, York YO10 5DD, UK.*

*Tel.: +44-(0)1904-432779 Fax: +44-(0)1904-432767 Email: Alan.Burns@cs.york.ac.uk*

## 1. Introduction

The HRG is concerned with the use of Ada in high integrity applications. Both to see how best to utilise Ada-based technology and to improve the quality of software engineering by the use of the best available programming language for this domain. The initial work of the HRG has been concerned with the production of Guidelines on the use of Ada. This work has now completed with the production of an ISO technical report [1]. More recently the HRG has concerned itself with the status of Annex H in the ARM, and with the use of the Ravenscar profile in high integrity applications. This report concerns the latter of these issues and represents the work of all members of the HRG. The currently expectation is that this work will also become an ISO technical report.

### 1.1 Motivation

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 deadlines can result in a catastrophic failure of the whole system. The choice of a suitable design and development methodology, 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.

Traditional methods used for the design and development of complex critical applications, which concentrate primarily on functionality, are inadequate for hard real-time systems. This is because non-functional requirements such as dependability, timeliness, memory usage and dynamic change management are left until too late in the development cycle.

The use of Ada has been proven to be of great value within high integrity and real-time applications, although it has been necessary to define a subset of deterministic constructs to ensure full analysability of the code. Such subsets have been defined for Ada83, but these have excluded Ada tasking on the grounds of its non-determinism and inefficiency. However, there have been recent advances in the area of schedulability analysis of task sets to ensure that all hard deadlines are met, even in the presence of a run-time system that enforces preemptive task scheduling based on multiple priorities. This valuable research work has been mapped to a number of new Ada constructs and rules that have been incorporated into the

Real-Time Annex of the Ada95 language standard. This has opened the way to support for these Ada tasking constructs in high integrity subsets whilst still retaining the core elements of predictability and reliability.

The Ravenscar Profile is a restricted Ada95 tasking model 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. The model is consistent with the use of tools that allow the schedulability of a system to be analysed and simulated throughout the development life-cycle, thus avoiding the common problem of finding only during system integration and testing that the design fails to meet its timing requirements.

In summary, the major goals that steered the definition of the Ravenscar Profile are:

- a) A deterministic tasking model to support current schedulability analysis theory to ensure that a task set meets its hard deadlines.
- b) A compact and highly efficient Ada run-time system suited to meeting tight memory constraints and performance requirements, and which could be certifiable to the highest integrity levels.

The traditional approach to formal verification and certifiability has been to dispense entirely with separate processes, each with their own independent thread of control, and to use a *cyclic executive* that calls each subtask as a procedure in turn in a fully deterministic manner. Such a system becomes easy to analyse, but is typically difficult to design, inflexible to change, and not well suited to applications where sporadic activity may occur and where error recovery is important. Moreover it can lead to poor software engineering if small procedures have to be artificially constructed.

The Ravenscar profile has been designed such that the restricted form of Ada tasking that it defines can be used even for software that needs to be verified to the very highest integrity levels. The profile has been already been included in an ISO technical report that provides guidelines to the use of Ada for high integrity systems. The aim of this guide is to give a complete description of the motivations behind the Profile, to show how conformant programs can be analysed and to give examples of usage.



## 1.2 Structure of the Report

The report will contain a number of chapters:

- Introduction
- Definition of Ravenscar
- Rationale for the features of Ravenscar
- Verification of Ravenscar compliant programs
- Code examples

In this paper we give only the Definition and Rationale. The reader should note that the work reported in this paper is only in draft form. The HRG is yet to finalise these, and the other sections. It is envisaged that the final report will be completed during 2002.

## 2. Ravenscar Profile Definition

The definition of the Profile will be incorporated into the Ada Reference Manual (ARM) in the following form.

An application requests the use of the Ravenscar profile by use of the `Profile pragma` with the identifier `Ravenscar`. The task dispatching policy that is required by `pragma Profile (Ravenscar)` is `FIFO_Within_Priorities`. A conforming implementation of the Profile may define alternative task dispatching policies, for example one that is non-preemptive, but in this case, the implementation must define an alternative identifier for `pragma Profile`, for example `pragma Profile (Ravenscar_Non_Preemptive)`. The locking policy that is required by `pragma Profile (Ravenscar)` is `Ceiling_Locking`.

### 2.1 Summary of Implications of pragma Profile (Ravenscar)

The following restrictions apply to the alternative mode of operation defined by the Ravenscar Profile. The first set come from Ada95's definition of restrictions:

```
Restrictions
    (Max_Asynchronous_Select_Nesting=0)
Restrictions    (Max_Protected_Entries=1)
Restrictions    (Max_Task_Entries=0)
Restrictions    (No_Abort_Statements)
Restrictions    (No_Asynchronous_Control)
Restrictions    (No_Dynamic_Priorities)
Restrictions    (No_Implicit_Heap_Allocations)
Restrictions    (No_Task_Allocators)
Restrictions    (No_Task_Hierarchy)
```

The next ones are newly defined:

```
Restrictions    (Max_Entry_Queue_Length=1)
Restrictions    (No_Calendar)
Restrictions    (No_Dynamic_Attachment)
Restrictions    (No_Local_Protected_Objects)
Restrictions    (No_Protected_Type_Allocators)
Restrictions    (No_Relative_Delay)
Restrictions    (No_Requeue_Statements)
Restrictions    (No_Select_Statements)
Restrictions    (No_Task_Attributes_Package)
Restrictions    (Simple_Barriers)
```

Finally there is one optional setting:

```
Restrictions    (Max_Tasks=N)
```

## 3 Rationale

This chapter provides a detailed rationale for the imposition of each restriction and some general discussion about how to work within the restrictions while still retaining flexibility in the design and coding processes.

### 3.1 Ravenscar Profile Restrictions

#### 3.1.1 Static existence model

The restrictions listed below ensure that the task and interrupt set to be analysed for schedulability is fixed and has static properties (in particular, base priority) after program elaboration. If a variable task set were to exist, then it would not be practically possible to perform static timing analysis of the program because of the dynamic nature of the requirements for CPU time and the meeting of deadlines.

#### No\_Task\_Hierarchy

*All (nonenvironment) tasks depend directly on the environment task of the partition.*

The restriction `No_Task_Hierarchy` prevents the declaration of tasks local to procedures or to other tasks. Thus tasks may only be created at the library level, i.e., within the declarative part of library level package specifications and bodies, including child packages and package subunits.

#### No\_Task\_Allocators

*There are no allocators for task types or types containing task subcomponents.*

The restriction `No_Task_Allocators` prevents the dynamic creation of tasks via the execution of Ada allocators.

#### No\_Abort\_Statements

*There are no abort statements, and there are no calls to Task\_Identification.Abort\_Task.*

The restriction `No_Abort_Statements` ensures that tasks cannot be aborted. The removal of abort statements (and select then abort) significantly reduces the size and complexity of the run-time system.

#### No\_Dynamic\_Attachment

*There is no call to any of the operations defined in package Ada.Interrupts (Is\_Reserved, Is\_Attached, Current\_Handler, Attach\_Handler, Exchange\_Handler, Detach\_Handler)*

The restriction `No_Dynamic_Attachment` excludes use of the operations in predefined package `Ada.Interrupts`, which contains primitives to attach and detach handlers dynamically during program execution. In conjunction with restriction `No_Local_Protected_Objects`, this implies that interrupt handlers can only be attached statically by

use of pragma `Attach_Handler` applying to protected procedures within library-level protected objects.

#### **No\_Dynamic\_Priorities**

*There are no semantic dependencies on the package `Dynamic_Priorities`.*

The restriction `No_Dynamic_Priorities` prevents use of the predefined package `Ada.Dynamic_Priorities`, thereby ensuring that the priority assigned at task creation is unchanged during the task's execution, except when the task is executing a protected operation, during which time it inherits the ceiling priority.

If static timing analysis is not of interest to the application program and a more general model of tasks and interrupts is required, this can still be achieved with reasonable expressive power within the subset definition.

Dynamic creation and termination of tasks can be simulated by declaring a pool of event response tasks at program startup, each containing an infinite loop which has a *suspending operation* as its first statement, such that its execution can be invoked dynamically by one of the task synchronization primitives. Thus, by changing the settings of suspension objects and entry barriers, it is possible for certain tasks to have their execution disabled whilst others have execution enabled.

Dynamic exchange of interrupt handlers, often required for applications performing *mode change*, can be simulated by embodying all the different handling code for a particular interrupt in one interrupt handler protected procedure, with each of the different actions being coded as case alternatives in a case statement, dependent on a mode selector. By changing the value of the mode selector, the same handler procedure can perform different response actions at various times during program execution.

Dynamic task priority change is also generally associated with mode change. This can be simulated by use of a separate event response task for each mode of operation (and assigning a different priority to each task as required), such that the execution of each task that belongs to a dormant mode is suspended until signalled when its mode becomes active.

### **3.1.2 Static synchronization and communication model**

These restrictions are a natural consequence of the static tasking model, since a locally declared or allocated protected object is meaningless for mutual exclusion and task synchronization purposes if it can only be accessed by one task. Furthermore, a static set of protected objects is required for schedulability analysis.

#### **No\_Local\_Protected\_Objects**

[New] *All protected objects are created via library-level declarations.*

The restriction `No_Local_Protected_Objects` prevents the declaration of protected objects local to subprograms, tasks, or other protected objects.

#### **No\_Protected\_Type\_Allocators**

*There are no allocators for protected types or types containing protected subcomponents.*

The restriction `No_Protected_Type_Allocators` prevents the dynamic creation of protected objects via Ada allocators.

#### **No\_Select\_Statements**

*There are no select statements.*

#### **Max\_Task\_Entries = N**

*Specifies the maximum number of entries per task.*

For the Ravenscar Profile, the value of `Max_Task_Entries` is zero.

The restrictions `Max_Task_Entries=0` and `No_Select_Statements` prohibit the use of Ada rendezvous for task synchronization and communication. This ensures that these operations are achieved using only the two supported task synchronization primitives: protected object entries and suspension objects, which both exhibit time-deterministic execution properties needed for static timing analysis.

If static timing analysis is not of interest to the application program, the classic non-timed rendezvous operations can still be achieved within the subset definition by use of suspension objects for synchronization and protected object entries for communication.

Note that no timed or conditional form of suspension is supported by the subset; this can be simulated if a suspension object is used by polling the state of the suspension object (via the `Current_State` function in package `Ada.Synchronous_Task_Control`), or if a protected entry is used by polling the value of the protected data which controls the synchronization (i.e., the barrier Boolean).

### **3.1.3 Deterministic Memory Usage**

The profile contains two restrictions that are designed to prevent implicit dynamic memory allocation by the implementation. Note that the profile does not prevent the use of the standard storage pool or a user-defined storage pool via explicit allocators, although if there were no application-level visibility or control over how the storage in the standard storage pool was managed, the use of this pool would not be recommended.

#### **No\_Implicit\_Heap\_Allocations**

*There are no operations that implicitly require heap storage allocation to be performed by the implementation. The operations that implicitly require heap storage allocation are implementation defined.*

The restriction `No_Implicit_Heap_Allocations` prevents the implementation from allocating memory

from the standard storage pool other than as part of the execution of an Ada allocator-.

#### **No\_Task\_Attributes\_Package**

*There are no semantic dependencies on the package Ada.Task\_Attributes.*

The restriction No\_Task\_Attributes\_Package prevents use of the predefined package Ada.Task\_Attributes, which is used to dynamically create attributes of each task in the application. Attribute creation generally implies hidden dynamic allocation of memory.

### **3.1.4 Deterministic execution model**

The following restrictions are present to provide deterministic execution:

#### **Max\_Protected\_Entries = N**

*Specifies the maximum number of entries per protected type. The bounds of every entry family of a protected unit shall be static, or shall be defined by a discriminant of a subtype whose corresponding bound is static.*

For the Ravenscar Profile, the value of Max\_Protected\_Entries is 1.

#### **Max\_Entry\_Queue\_Length = N**

*Specifies the maximum number of calls that may be queued concurrently on an entry.*

For the Ravenscar Profile, the value of Max\_Entry\_Queue\_Length is 1, and a call can only be queued on a protected entry, since Max\_Task\_Entries is zero.

The restrictions Max\_Protected\_Entries=1 and Max\_Entry\_Queue\_Length=1 ensure that at most one task can be suspended waiting on a closed entry barrier for each protected object which is used as a task synchronization primitive. This avoids the possibility of queues of tasks forming on an entry, with the associated non-determinism of the length of the waiting time in the queue. It also avoids the possibility of there being more than one barrier becoming open simultaneously as the result of a protected action, with the associated non-determinism of selecting which entry should be serviced first.

Note that the Max\_Entry\_Queue\_Length restriction is checked at run time and so violation results in the raising of Program\_Error exception at the point of the entry call. This is consistent with the Ada rule that states that Program\_Error exception is raised upon calling Suspend\_Until\_True if another task is waiting on that suspension object

When the restriction Max\_Entry\_Queue\_Length = 1 is in force, pragma Queuing\_Policy has no effect, since there are no queues.

#### **Simple\_Barriers**

*The Boolean expression in an entry barrier shall be the value of a Boolean literal or a Boolean component of the enclosing protected object.*

The restriction Simple\_Barriers, coupled with Max\_Protected\_Entries=1, ensures a deterministic execution time and absence of side effects for the evaluation of entry barriers at the epilogue of protected actions within a protected object that is used for task synchronization. There is also scope for additional optimization by the implementation since the barrier value is either static or can be read directly from one of the protected object components, without needing to be computed separately. If the application requires composite entry barrier expressions, this can be achieved by declaring an additional Boolean in the protected data and assigning the composite expression to the Boolean whenever its evaluation result may change.

#### **No\_Requeue\_Statements**

*Requeue statements are not allowed.*

The restriction No\_Requeue\_Statements ensures deterministic task release from protected entry barriers used for task synchronization. The requeue statement in Ada causes the current caller of a protected entry to be requeued to a different entry dynamically, thereby making it difficult to perform static analysis of task release. A similar effect to requeue can be achieved by completing the protected entry body and returning a status result to the caller, who can then emit a subsequent protected entry call to the intended destination of the requeue statement. If each protected entry is called only by a single task, then this alternative technique does not introduce any race conditions.

#### **Max\_Asynchronous\_Select\_Nesting = N**

*Specifies the maximum dynamic nesting level of asynchronous\_selects. A value of zero prevents the use of any asynchronous\_select.*

For the Ravenscar Profile, the value of Max\_Asynchronous\_Select\_Nesting is zero. This excludes the use of asynchronous transfer of control. This ensures that task execution is functionally deterministic.

#### **No\_Asynchronous\_Control**

*There are no semantic dependencies on the package Asynchronous\_Task\_Control.*

The restriction No\_Asynchronous\_Control excludes the use of asynchronous suspension of execution. This ensures that task execution is temporally deterministic.

#### **No\_Relative\_Delay**

*Delay\_relative statements are not allowed.*

The restriction No\_Relative\_Delay prohibits use of the delay\_relative statement based on type Duration. This statement exhibits non-determinism with respect to the absolute time at which the delay expires in the case when the delaying task is preempted after calculating the required relative delay, but before actual suspension occurs. In contrast, the

deterministic delay\_absolute statement **delay until** should be used instead for accurate release of cyclic tasks.

### No\_Calendar

*There are no semantic dependencies on the package Ada.Calendar.*

The restriction No\_Calendar ensures that all timing is performed using the high precision afforded by the time type in package Ada.Real\_Time, or by an implementation-defined time type. The Ada.Real\_Time time type has a precision of the same order of magnitude as the real-time clock device on the underlying board. In contrast, the time type in package Calendar generally has much coarser precision than the real-time clock, due to the need to support a 200 year range, and so its use would result in inaccuracy in task release times.

### 3.1.5 Optional restrictions

#### Max\_Tasks = N

*Specifies the maximum number of task creations that may be executed over the life time of a partition, not counting the creation of the environment task.*

The restriction Max\_Tasks=N is not a mandatory restriction of the Ravenscar Profile and is not implied by pragma Profile (Ravenscar). The Max\_Tasks pragma supplies optional additional information to the implementation on the maximum number of tasks that can be created in the partition. An implementation that conforms to the Ravenscar Profile need not enforce this restriction, but the recommendation is that a conforming implementation should at least enforce Max\_Tasks=0, i.e. this can be used as a predicate to assert that the application program does not use Ada tasking at all.

## 3.2 Ravenscar Profile Dynamic Semantics

### 3.2.1 Task Dispatching Policy

The task dispatching policy that is required by pragma Profile (Ravenscar) is FIFO\_Within\_Priorities. A conforming implementation of the Profile may define alternative task dispatching policies, for example one that is non-preemptive (i.e. “run until block”), but in this case, the implementation must define an alternative identifier for pragma Profile, for example pragma Profile (Ravenscar\_Non\_Preemptive).

A non-preemption implementation of the Profile for a mono-processor is in between the cyclic executive approach and the preemptive tasking approach with regard to ease of timing analysis, flexibility with regard to change, and responsiveness to asynchronous events. In common with the cyclic executive approach, there is no contention for shared resources, and there is no need to analyse the impact from asynchronous events, but there is still the need to break up long code sequences using voluntary suspension points to obtain reasonable responsiveness to asynchronous events. In common with the preemptive

scheduling approach, the order of execution of tasks is affected by responses to asynchronous events due to being priority based.

### 3.2.2 Locking Policy

The locking policy that is required by pragma Profile (Ravenscar) is Ceiling\_Locking. This policy provides one of the lowest worst case blocking times for contention for shared resources, and so maximizes the schedulability of the task set when preemptive scheduling is used.

For a non-preemption implementation of the Profile for a mono-processor, no locking of protected objects is required.

### 3.2.3 Additional Run time Errors Defined by the Ravenscar Profile

All except one of the pragma Restrictions that are implied by pragma Profile (Ravenscar) are enforced at compile-time, and violation of the restriction will result in a compiler-generated error message. The exception is Max\_Entry\_Queue\_Length=1, which can only be detected at run-time unless further restrictions are imposed on the calling style for protected entries to allow program-wide static analysis to ensure that at most one task calls each entry. These further restrictions are not required of a conformant implementation of the Profile – it is sufficient to detect a violation of this error at run-time and to raise Program\_Error at the point of the entry call (as is required for the corresponding call to Suspend\_Until\_True for a suspension object).

Furthermore, conformance to the Ravenscar Profile requires an implementation to detect the use of the following potentially blocking operations within a protected action, and hence to raise Program\_Error exception:

- a protected entry\_call\_statement
- a delay\_until\_statement
- a call to a language-defined subprogram that is potentially blocking, for example Ada.Synchronous\_Task\_Control.Suspend\_Until\_True
- a call to a subprogram that contains one of the above potentially blocking operations

This requirement is to allow conforming implementations to implement Ceiling\_Locking on a mono-processor using ceiling priority alone as the mutual exclusion lock. This implementation is very efficient and temporally deterministic. In particular, this is a non-queueing implementation of the lock. On a multi-processor, a mutual exclusion (*mutex*) lock must be used and so queues can form to gain access to the protected object. In this case, the worst case blocking time can still be computed based on the fact that each holder of the lock in turn will execute at ceiling priority, assuming a FIFO queueing policy for the mutex queue.

Note that the Ravenscar Profile subset excludes the following potentially blocking operations within protected operations:

- a select statement
- an accept statement
- a task entry call
- a relative delay statement
- an abort statement
- task creation or activation
- an external requeue statement with the same target object as that of the protected action.
- Note also that the Profile does not require detection of the potentially blocking operations that are in:
- an external subprogram call with the same target object as that of the protected action

- a call to an implementation-defined subprogram that is potentially blocking, for example one in which the blocking occurs within the execution of an Imported subprogram or other foreign language code.

#### **4. Conclusion**

Ada remains the most effective language for the production of high integrity systems. The HRG attempts to affirm and support this position by producing guidelines on the use of Ada. Its current focus is on the Ravenscar Profile which is proving to be an important piece of Ada technology. During the coming year the HRG will produce guidelines for the use of Ravenscar. This paper represents an early draft of part of this report.

#### **References**

- [1] Guide for the use of Ada programming language in high integrity system, ISO/IEC Technical Report 15942, 2000

# Ada-Europe Associate Members (National Ada Organizations)

## Ada-Belgium

attn. Dirk Craeynest  
c/o Offis nv/sa  
Weiveldlaan 41/B32  
B-1930 Zaventem  
Belgium

Phone: +32-2-725-40-25  
Fax: +32-2-725-40-12  
Email: Dirk.Craeynest@offis.be  
URL: [www.cs.kuleuven.ac.be/~dirk/ada-belgium](http://www.cs.kuleuven.ac.be/~dirk/ada-belgium)

## Ada in Denmark

attn. Jorgen Bundgaard  
c/o DDC-I  
Gl. Lundtoftevej 1B  
DK-2800 Lyngby  
Denmark

Phone: +45-45-871144  
Fax: +45-45-872217  
Email: [jb@ddci.dk](mailto:jb@ddci.dk)

## Ada-Deutschland

attn. Dr. Peter Dencker  
Aonix GmbH  
Durlacher Allee 95  
D-76137 Karlsruhe  
Deutschland

Phone: +49-721-98653-22  
Fax: +49-721-98653-98  
Email: [dencker@aonix.de](mailto:dencker@aonix.de)  
URL: [ada-deutschland.de](http://ada-deutschland.de)

## Ada-France

chez Fabrice Kordon  
48 rue Vergniaud  
F-75013 Paris  
France

Phone: +33-1-44 27 61 89  
Fax: +33-1-44 27 62 86  
Email: [bureau@ada-france.org](mailto:bureau@ada-france.org)  
URL: [www.ada-france.org](http://www.ada-france.org)

## Ada-Spain

attn. Francisco Perez-Zarza  
P.O. Box 50.403  
E-28080 Madrid  
Spain

Phone: +34-1-627-8247  
Fax: +34-1-309-3685  
Email: [fperez@ceselsa.es](mailto:fperez@ceselsa.es)  
URL: [www.adaspain.org](http://www.adaspain.org)

## Ada in Sweden

Ada I Sverige  
c/o Mariadata  
Box 1085  
SE-141 22 Huddinge  
Sweden

Phone: +46-08-779-88-30  
Fax: +46-08-774-37-93  
Email: [info@ada-i-sverige.se](mailto:info@ada-i-sverige.se)  
URL: [www.ada-i-sverige.se](http://www.ada-i-sverige.se)

## Ada –Switzerland

attn. Henri Roethlisberger  
EINEV, Ecole d'Ingenieurs de l'Etat de Vaud  
CP 587  
CH-1401 Yverdon-les-Bains  
Switzerland

Phone: +41-24-423-22-63  
Fax: +41-24-425-00-50  
Email: [roethlisberger@einev.ch](mailto:roethlisberger@einev.ch)

## Ada Language UK

attn. Helen Byard  
P.O. Box 322  
York YO10 3GY  
UK

Phone: +44-(0)1904-412740  
Fax: +44-(0)1904-426702  
Email: [admin@adauk.org.uk](mailto:admin@adauk.org.uk)  
URL: [www.adauk.org.uk](http://www.adauk.org.uk)

# Ada UK 2001 Sponsors

## **ACT Europe**

Contact: *Franco Gasperoni*

8, Rue de Milan, 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

## **Alenia Marconi Systems**

Contact: *Don Harvey*

Eastwood House, Glebe Rd., Chelmsford, Essex, CM1 1QW, UK  
Tel: +44-(0)1276-696901  
Email: info@aonix.co.uk

Fax: +44-(0)1276-659842  
URL: www.aonix.co.uk

## **Aonix Europe Ltd**

Contact: *Neil Michniak*

Partridge House, Newtown Rd., Henley on Thames, Oxon, RG9 1HG, UK  
Tel: +44-(0)14941-415000  
Email: info@aonix.co.uk

Fax: +44-(0)14941-571866  
URL: www.aonix.com

## **ARTISAN Software Tools**

Contact: *Peter Kibble*

Stamford House, Regent St., Cheltenham, Glos., GL50 1HN, UK  
Tel: +44-(0)1242-229320  
Email: peterk@artisansw.com

Fax: +44-(0)1242-229301  
URL: www.artisansw.com

## **BAE SYSTEMS**

Contact: *Paul McCormack*

Warwick House, PO Box 87, Farnborough Aerospace Centre, Farnborough, Hants, GU14 6YU, UK  
Email: Paul.McCormack@baesystems.com

URL: www.baesystems.com

## **Data Systems and Solutions**

Contact: *Dave Woodhall*

SEAS Building, Sinfine Lane, Derby, DE24 8BJ, UK  
Tel: +44-(0)1332-771700  
Email: info@ds-s.com

Fax: +44-(0)1332-770921  
URL: www.ds-s.com

## **EDS**

Contact: *Lee Edwards*

Hartley House, 15 Bartley Wood Business Park, Bartley Way, Hook, Hants., RG27 9XA, UK  
Tel: +44-(0)1256-741122  
Email: swep.sales@eds.com

Fax: +44-(0)1256-741132

## **First Matrix Ltd**

Contact: *Alan Barker*

Old Lion Court, High St., Marlborough, Wilts., SN8 1HQ., UK  
Tel: +44-(0)1672-515510  
Email: arb@ftmx.com

Fax: +44-(0)1672-515514

## **Green Hills Software Ltd**

Contact: *Jon Williams*

Chancery Court, Lincoln Rd., High Wycombe, Bucks., HP12 3RE., UK  
Tel: +44-(0)1844-267950  
Email: sales-uk@ghs.com

Fax: +44-(0)1844-267955  
URL: www.ghs.com

## **IPL Information Processing Ltd**

Contact: *Ian Gilchrist*

Eveleigh House, Grove St., Bath, BA1 5R., UK  
Tel: +44-(0)1225-475114  
Email: ipl@iplbath.com

Fax: +44-(0)1225-444400  
URL: www.iplbath.com

## **LDRA Ltd**

Contact: *Jim Kelly*

24 Newtown Rd., Newbury, Berks., RG14 7BN, UK  
Tel: +44-(0)635-528828  
Email: sales@ldra.com

Fax: +44-(0)635-528657  
URL: www.ldra.com

## **Objektum**

Contact: *Derek Russell or Ahmed Amin*

Units 2/3 Cranleigh Works, The Common, Cranleigh, GU6 8SB, UK  
Tel: +44-(0)1483-278178  
Email: info@objektum.com

Fax: +44-(0)1483-275384  
URL: www.objektum.com

## **Praxis Critical Systems Ltd**

Contact: *Peter Amey*

20 Manvers St., Bath, BA1 1PX, UK  
Tel: +44-(0)1225-469991  
Email: sparkinfo@praxis-cs.co.uk

Fax: +44-(0)1225-469006  
URL: www.praxis-cs.co.uk

## **Rational Software Ltd**

Contact: *Roger Bowser*

Kingswood, Kings Ride, Ascot, Berks., SL5 8AJ, UK  
Tel: +44-(0)1344-295000  
Email: info@rational.com

Fax: +44-(0)1344-295001  
URL: www.rational.com

## **John Robinson & Associates**

Contact: *John Robinson*

2 Currer St., Oakenshaw, Bradford, W. Yorks., BD12 7DP, UK  
Tel: +44-(0)1274-691935  
Email: John@jr-and-assoc.demon.co.uk

Fax: +44-(0)8700-558750  
URL: www.jr-and-assoc.demon.co.uk

## **Telelogic UK Ltd**

Contact:

Chancery House, 8 Edward St., Birmingham, B1 2RX, UK  
Tel: +44-(0)121-2346600  
Email: info@telelogic.com

Fax: +44-(0)121-2346611  
URL: www.telelogic.com

## **TNI Europe Ltd**

Contact: *Tony Elliston*

58a Mill St., Congleton, Cheshire, CW12 1AG, UK  
Tel: +44-(0)1260-291449  
Email: info@tni-europe.com

Fax: +44-(0)1260-291449  
URL: www.tni-europe.com

## **Wind River Systems UK Ltd**

Contact: *David Bew*

Unit 5 & 6, 1<sup>st</sup> Floor, Ashted Lock Way, Aston Science Park, Birmingham, B7 4AZ, UK  
Tel: +44-(0)121-3590999  
Email: inquiries-uk@windriver.com

Fax: +44-(0)121-3804444  
URL: www.windriver.com