Contents

Editorial Policy for *Ada User Journal*

Publication

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

Aims

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

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

The Journal publishes the following types of material:

- Refereed original articles on technical matters concerning Ada and related topics.

- News and miscellany of interest to the Ada community.

- Reprints of articles published elsewhere that deserve a wider audience.

- Commentaries on matters relating to Ada and software engineering.

- Announcements and reports of conferences and workshops.

- Reviews of publications in the field of software engineering.

- Announcements regarding standards concerning Ada.

Further details on our approach to these are given below.

Original Papers

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

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

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

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

Unless the authors state explicitly otherwise, submission of an article is taken to imply that it represents original, unpublished work, not under consideration for publication elsewhere.

News and Product Announcements

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

Reprinted Articles

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

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

Commentaries

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

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

Announcements and Reports

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

Reviews

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

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

Submission Guidelines

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

Prospective authors are encouraged to contact the Editor by email to determine the best format for submission. Contact details can be found near the front of each edition.

Example papers conforming to formatting requirements as well as some word processor templates are available from the editor. There is no limitation on the length of papers, though a paper longer than 10,000 words would be regarded as exceptional.

Editorial

With this issue, I am taking over from Neil Audsley as Editor-in-Chief of the Ada User Journal. I wish to thank the Ada-Europe board for graciously offering me the job, which I hope will be as rewarding to me and to the readership as it is, no doubt, demanding.

At this time I want to especially greet all Journal readers and subscribers: I look forward to a satisfactory double-ended relationship. On my part, I commit to helping the Journal continue to disseminate valuable information, both specific to the Ada community and more generally to the ever-moving world of software engineering. In return, I expect the readers will also want to be inspired contributors of any sort of material that fits the scope and aims of the Journal.

One immediate objective is to bring the actual date of issue of the Journal back to the intended schedule: this will preserve and possibly increase the face value of the Journal. For as obvious as it may sound, however, this goal is a hard one to attain, for preparing and editing an issue currently is quite a job for (busy) volunteers. Please look with sympathy at this effort.

The News section forms the prominent part of this issue: it provides a vivid representation of how Ada permeates the professional, educational and intellectual activity of its user community.

While we are striving to build a long-ranging backlog for the Articles section (for which you contributions are eagerly sought!), this issue features a paper by active Ada users in Germany, which shows how Ada and UML can form a powerful combination for the development of distributed, reactive software systems.

The Conference Calendar section completes the offer of this issue.

In closing this editorial, I remind all readers that the Journal is theirs: let us together make it increasingly better!

Tullio Vardanega

Padova

September 2002

Email: *tullio.vardanega@math.unipd.it*

News

Dirk Craeynest (ed)

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

Contents

Ada-related Organizations

ARG Approves AI on Handling Mutually Recursive Types

*From: Randy Brukardt
  <randy@rrsoftware.com>*
Date: Fri, 28 Jun 2002
Subject: Re: OOD in Ada? Correction
Newsgroups: comp.lang.ada

There are now 4 different proposals on how to solve this problem (AI95-00217-01-04). So I am afraid we still have to wait a while until we get some working solution to this problem.

Well, the good news is that while this was being debated here, the ARG was meeting in Vienna [at the Ada-Europe'2003 conference -- dc]. And we approved (to my great surprise) AI-00217-04. So we have a "final" solution. How long it will be before it starts showing up in compilers I don't know (several vendors have worked on it, but whether it will be available in their compilers soon I can't say).

[http://www.ada-auth.org/cgi-bin/cvsweb.cgi/AIs/AI-30217.TXT -- dc]

Randy Brukardt, ARG Editor

Ada-related Events

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

Jun 21 - Ada-Europe Workshop on a Standard Container Library for Ada

*From: dirk@cs.kuleuven.ac.be (Dirk
  Craeynest)*
Date: 22 May 2002
*Subject: Standard Container Library for
  Ada, CfP Ada-Europe'2002 workshop*
*Newsgroups:
  comp.lang.ada,fr.comp.lang.ada*

[At this years' Ada-Europe'2002 conference in Vienna, Austria, a "Standard Container Library for Ada" workshop was held on June 21. -- dc]

http://www.ada-europe.org/conference2002.html
->Program->Workshop

[...] Both contemporary dominant general purpose programming languages, Java and C++, come equipped with a standard set of reusable containers, such as Maps and Sets. There are quite a few Ada libraries for these purposes, but there is little agreement on the exact details of a standard container library. There is however a general feeling, as can be witnessed on recent discussions on *comp.lang.ada*, that such a library is important for Ada's future.

[...] The submissions and a workshop summary will be published in the Ada User Journal. [...]

Suggested programming problems for evaluating data structure libraries:
http://www.auto.tuwien.ac.at/AE2002/problems.html

Workshop Proposals and other Resources:
http://www.auto.tuwien.ac.at/AE2002/resources.html [...]

Workshop co-chairs: John English (U. of Brighton) & Ehud Lamm (Open U. of Israel) [...]

Dirk.Craeynest@cs.kuleuven.ac.be
Ada-Europe'2002 Publicity Chair

Jun 25 - Ada-Belgium Event on Automatic Analysis and Manipulation of Source Code

*From: dirk@cs.kuleuven.ac.be (Dirk
  Craeynest)*
Date: 23 Jun 2002
*Subject: RainCode Technology Update -
  REMINDER*
*Newsgroups:
  comp.lang.ada,fr.comp.lang.ada,be.comp.
  programming,nl.comp.programmeren*

Ada-Belgium is pleased to announce a technical presentation by Jean-Christophe Real of RainCode in Brussels on "Automatic Analysis and Manipulation of Source code"

Tuesday, June 25, 2002, 20:00-21:30, at Offis nv/sa - Aubay Group, Gatti de Gamondstraat 145, B-1180 Brussels

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

Description of the presentation:

The RainCode technology: what it is, how it works, what it can be used for, etc.

RainCode's three main products: (1) The RainCode Engine performs automatic user definable code analysis, inventory, and transformation of source code in Ada, APS, C & C++, COBOL, CSP, Delphi, Ideal, Informix 4GL, Java, Natural, and PL/1. (2) The RainCode Roadmap automatically generates complete technical documentation of Ada sources. It enables quick impact analysis and gives a number of quality metrics about the code. The programmer can navigate very easily through the code thanks to this tool. (3) The RainCode Checker verifies compliance to a number of pre-defined Ada standards.

We will show you how to code some sample programs of the RainCode Engine for Ada to find the global variables, for instance, or apply specific transformations in the code.

We will also show you how RainCode can be used to check for compliance to coding standards.

More information on RainCode:
http://www.raincode.com/

[...] Dirk Craeynest, Ada-Belgium Board,
ada-belgium-board@cs.kuleuven.ac.be

Oct 29 - Ada UK User Group Autumn Conference

*From: Hazel (Adaxia)
  <hazel@adaxia.com>*
Date: Wed, 28 Aug 2002
*Subject: Promotion of the Ada UK User
  Group Autumn Conference*
*To:Dirk Craeynest
  <Dirk.Craeynest@cs.kuleuven.ac.be>*

Embedded Systems Club and Ada UK User Group Autumn Conference.

Our autumn conference promises to be our best event yet, with 22 sessions in 3streams. The event is to be held in Swindon, UK on 29th October 2002 and is open to all:

[...] The event includes: a dedicated testing stream; a dedicated Ada stream; tutorials on "Test-First Design" and "Principles of High Availability Embedded Systems Design"; the opportunity to learn about, and contribute to, the development of new standards; a session dedicated to Ada code generation from UML models; a pre-event survey and birds-of-a-feather session on recruitment and retention of Ada staff; technical presentations on many embedded systems topics, including "C++ in Embedded Systems", "Automated software design/code verification" and "Scenario Based Testing".

[...] From the outset we have planned this conference to provide more than a sequence of presentations to sit and listen to. There are many opportunities to actively contribute to the event, including: a pre-event survey and birds-of-a-feather session on staff recruitment and retention; two panel sessions, one dedicated to a proposal to develop a standardised Ada code generation profile for the UML.

In addition, we have a wealth of sessions providing training and education on tools and techniques including two tutorials on Test-First Design (a key technique in eXtreme Programming) and Principles of High Availability Embedded Systems Design.

[...] Registrations can be made via the websites:

http://www.EmbeddedSystemsClub.com
http://www.AdaUK.org.uk

[...] John Robinson, Conference Director, John@JohnRobinsonAndAssociates.com

Dec 8-12 - ACM SIGAda 2002 Conference

From: Currie Colket <colket@mitre.org>
Date: Mon, 10 Jun 2002
Organization: The MITRE Corporation
Subject: Invitation to Participate in SIGAda 2002 at Houston At Clear Lake, Texas, 8-12 December
Newsgroups: comp.lang.ada

SIGAda will be conducting its SIGAda 2002 Conference in Houston at Clear Lake, Texas this December from 8-12 December. Houston has been an excellent technology-based city and is home to many NASA activities. Our planned hotel venue will be right across the street from the NASA/Johnson Space Center. We are currently working with NASA for a tour of their facilities as part of our evening activities.

This is an excellent venue for our Ada/software engineering conference. [...] We look forward to seeing you in sunny Texas! This is a beautiful time to go to Texas as the weather is extremely pleasing and beaches are still pleasant.

[...] More information can be found at the SIGAda 2002 Home Page at:

http://www.acm.org/sigada/conf/sigada2002

[Cf. also further in this Ada User Journal issue. -- dc]

Currie Colket, Chair ACM SIGAda

For any questions, please contact Dr. John McCormick, the Program Chair, at McCormick@cs.uni.edu or myself at colket@acm.org [...]

Ada Semantic Interface Specification (ASIS)

ASIS Documentation

From: Stephen Leake
<stephen.a.leake.1@gsfc.nasa.gov>
Date: 01 Jun 2002
Organization: NASA Goddard Space Flight Center (skates.gsfc.nasa.gov)
Subject: Re: ASIS
Newsgroups: comp.lang.ada

I am pretty new to Ada and I need to find some good, complete references to the Ada Semantic Interface Specification. Can anyone point me to a book that deals with it?

I'm not aware of any books on ASIS. The source code is actually the best documentation. Once you get used to it, it's not too bad.

There's an ASIS mailing list. See the SigAda ASIS web page for more info:

http://info.acm.org/sigada/wg/asiswg/asiswg.html

From: Colin_Paul_Gloster@acm.org
Date: 03 Jul 2002
Organization: Dublin City University
Subject: Re: Asis Queries and Element_LIsts
Newsgroups: comp.lang.ada

Perhaps if you check the ASIS documentation (270 Francs from ISO):

http://www.iso.ch/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=27169&ICS1=35&ICS2=60&ICS3=

the functionality you require may be found. Otherwise you could join and ask on the technical ASIS email list instead of asking on this Ada (as in Ada itself) forum.

In April and May, it was commented on the list <SIGAda-ASIS-tech@acm.org> that [...] it is advisable to consolidate libraries built on top of ASIS for higher level requests. Dubbed secondary libraries" you can find an explanation at:

http://www.acm.org/sigada/WG/asiswg/ASISWG_Results.html#Secondary_Library

[...] Some such libraries include [...]
http://www.acm.org/sigada/WG/asiswg/ASIS_Clients.html#Blake

Adasubst & Adadep - New Free Semantic Tools for Ada

From: Jean-Pierre Rosen
<rosen@adalog.fr>
Date: Thu, 1 Aug 2002
Organization: Adalog
Subject: Adasubst/Adadep: new free semantic tools for Ada released
Newsgroups: comp.lang.ada

Ever wanted to change that Bad_Name in a package spec for Better_Name?

Ever wanted to split that big package into several child units? But the amount of changes in units using the packages prevented you from doing it....

Adasubst is for you! It will do all the necessary changes, and since it is a *semantic* tool, it will account for things like use clauses and visibility rules, so only the correct elements will be changed, even in the presence of aliases.

Adadep is a companion utility, that tells you which elements from a package are actually used by one (or more) unit. A good hint in preparing a split of a package into child units!

These ASIS-based utilities are available from Adalog's components page
http://www.adalog.fr/compo2.htm
in source form and executables for Linux and Windows, with full HTML documentation.

They are released under the GMGPL, so you can not only use them, but reuse any part of them freely, including into your own proprietary applications.

Note to ASIS users: there is a 'Thick_Queries" package that you might be interested in, including a quite sophisticated "Full_Name_Image" function. Enjoy!

Ada and Education

Grabbing Mindshare in the Student Population for Ada

From: Richard Riehle
<richard@adaworks.com>
Date: Tue, 21 May 2002
Organization: AdaWorks Software Engineering
Subject: Re: Grabbing Mindshare in the Student Population for Ada
Newsgroups: comp.lang.ada

When I started teaching Ada at Naval Postgraduate School, the number of students enrolling for the class was so low it was not guaranteed there would be a class. With the help of a friendly professor who had been there for a long time, we were able to put together enough for one class. After that class, students began to tell other students about how much fun Ada was. We are ever so

gradually making progress at increasing mind-share, as the caption put it.

My class is called, Ada As A Second Language, so the students have almost all suffered through the horrors of C++ by the time they get to me. The sign on my office door says, "C++ Is Its Own Virus" and few of my students disagree with that sentiment. Just today, one of my students from the electrical engineering curriculum said how much he was enjoying Ada. I have even been able to persuade some students to use Ada for their Master's Degree Thesis.

At lunch today, with a group of Marines from one of my other classes, and one Marine visitor, the visitor mentioned how some project was being converted from Ada to Java. I said, "That's a pretty stupid decision." He asked, "What would be better?" I replied, "Ada 95." He said that everything he had heard about Ada was pretty negative. This is an indication that there is still a lot of ignorance out there in the decision loop.

We can, if we teach Ada well, grab some mind-share. However, if the students are confronted with negative attitudes when they try to use Ada (or recommend Ada) once they have graduated, it is pretty discouraging. On the bright side, some of my students will be in decision-making jobs when they graduate, and they may be able to help turn the tide of stupidity that characterizes so many programming language decisions.

If you can teach Ada well and help the students enjoy it, all the better. I am seeing the results of teaching it badly in so many places.

By the way, I expect to have an update of my little booklet, Ada Distilled, available for free download sometime in the next couple of weeks. I have been getting great feedback on it from all over the world. Apparently it has become useful for a lot of people who are trying to learn Ada on their own. I will announce it here when it was on adapower and adaic.org.

[Cf. also "Ada Distilled - Online Book Updated" further in this news section. -- dc]

From: byhoe@greenlime.com (Adrian Hoe)
Date: 22 May 2002
Subject: Re: Grabbing Mindshare in the
    Student Population for Ada
Newsgroups: comp.lang.ada

I was having negative feedback from a local university in Malaysia, UTM, the only university here that teach Ada. Mainly, the students can't see the benefits of Ada and are obfuscated by promotion and words of mouth. The lecturer came to me and asked for some suggestion how he can introduce Ada more successfully. Judging that his confidence had been shaken, I told him to stick to Ada and show his confident to his students and be more persistent. I also told him to provide

comparisons between Ada and other languages. He did it and his confidence is back again. :) He now asks students who are not convinced in Ada to compete with his students who learn Ada. And the result is great.

Adrian Hoe, http://adrianhoe.com

From: Chris Danx <danx@ntlworld.com>
Date: Thu, 23 May 2002
Subject: Re: Grabbing Mindshare in the
    Student Population for Ada
Newsgroups: comp.lang.ada

Everyone here knows there's more to programming than a GUI, but new newbies don't know that. Almost all new newbies see GUI programs and think it'd be cool to write apps like that. They take a course in a language and find they're writing console programs, which don't have the same appeal. If the concepts of programming and a bit of "shinyness" can be successfully integrated students might be more interested.

In our first year, the first thing our lecturer did was demonstrate a GUI based program written with the win32 binding. This seemed to grab the students attention, and it was maintained through the exercises they set for us. One involved creating a simple planetary system with a sun, planet and moon complete with shadows (all the exercises used Adagraph, not the win32 binding), which people had a lot of fun with (one guy had 9 planets and god knows how many moons working, just for the hell of it). Another involved plotting a bar chart of some data read from a file.

None of them was really about graphics or GUIs, they were about programming concepts but were presented in a way that made it fun and 'pleasing on the eye'. In second year they switched to textual programs, and the enthusiasm for programming seemed much less than it was in first year. Second year (Ada) programming courses are about data structures, algorithms and Software designs concepts (generics, OO...), but that doesn't mean they couldn't have come up with exercises that use graphical elements to keep interest (it just takes some imagination). Perhaps that might have offset the slightly increased difficulty of the exercises. Of course there is always the danger of students focusing more on the aesthetic quality of the program, but hopefully by making the graphical element relatively simple (and by getting tutors to keep an eye on students) that could be avoided (the graphical elements could also be weighted much less, like 1 (or 0) mark(s) for the whole exercise and 9 for the rest).

Just a thought.

From: tatebll@aol.com (Bill Tate)
Date: 23 May 2002
Subject: Re: Grabbing Mindshare in the
    Student Population for Ada

Newsgroups: comp.lang.ada

The posts above make a number of good suggestions but I would strongly advise backing up even earlier. I would point to the objectives of the Python community's CPE (Computer programming for everyone) initiative. It seeks to get people exposed to computer programming well before they reach college. If Ada is going to achieve "greater" mindshare (for all the right reasons), it would seem to me that you have to deal with the difficult problem of overcoming some really bad software development habits learned by programmers very early on, e.g., pre-college.

Ada strongly encourages greater discipline in the "engineering" of software; not the kind of attitude typically exhibited by those of the "hacker" and "design-by-keyboard" persuasion (to my great disappointment - I include no small number of CS graduates in this crowd). Mix that with a seemingly endless supply of individuals who believe themselves to possess "god-like" abilities when it comes to their software development prowess and everything points to the need for early intervention. Of course, there is the "cold turkey" approach of throwing 100K SLOC of C++ code in front of potential adopters and telling them to "maintain it." :>)

Lego Mindstorms

From: gab@rational.com (Greg Bek)
Date: 5 Jun 2002
Subject: Re: Embedded Ada Development
    Tools
Newsgroups: comp.lang.ada

One of these days, I'm going to buy my son the Lego Mindstorms kit and get that Ada compiler working.

The best way to do this is to start with Barry Fagin's ada2nqc program. It allows you to write pure Ada code and then translate it into NQC for compilation and download to the RCX.

I took Barry's code and added tasking and a couple of other things to it, although I think Barry has since done it as well. There is some slight mismatch between Ada tasking semantics and the Lego RCX tasking, but it doesn't stop you writing pure Ada code. The code might not work if ported to another target, but that isn't a real problem in this domain. [...]

From: Jerry Petrey
    <jdpetrey@raytheon.com>
Date: Thu, 06 Jun 2002
Organization: Raytheon Company
Subject: Re: Embedded Ada Development
    Tools
Newsgroups: comp.lang.ada

Here are some links to Barry's work on Ada Mindstorms:

http://www.faginfamily.net/barry/Papers/AdaLetters.htm

http://www.usafa.af.mil/dfcs/Ada_Mindstorms_m
anual.htm
http://www.faginfamily.net/barry/Papers/IEEERA
.htm
http://www.acm.org/sigada/conf/sigada2001/pri-
vate/SIGAda2001-CDROM/SIGAda2001-
Proceedings/Fagin-Paper.pdf

[Cf. also "Fun with Ada Lab at
AdaPower" in AUJ 22-4 (December
2001), p.205, and "An Embedded
Programming Course in a Bag?" in AUJ
22-2 (June 2001), pp.66-70. -- dc]

Jerry Petrey, Senior Principal Systems
Engineer, Raytheon Missile Systems

Course and Training Material in French

[Extracts translated from French: -- dc]

From: DamBec <dambec@hotmail.com>
Date: Wed, 5 Jun 2002
Subject: Re: Demande de conseil :
*   Programmation systeme*
Newsgroups: fr.comp.lang.ada

"Programmer en Ada 95" by John Barnes
(translated by Hugues Fauconnier),
edition Vuibert [...]

http://lithwww.epfl.ch/teaching/cmp/

[Web-page for the "Cours de
programmation Ada" at the Computer
Science Theory Laboratory (LITH) of the
EPFL in Switzerland; includes course
notes. -- dc]

From: Claude Kaiser <kaiser@cnam.fr>
Date: Thu, 4 Jul 2002
Subject: [ada-france] Diffusion de cours
*   systemes*
To: ada-france@ada-france.org

For distribution: If you are looking for a
systems course ready to use, with all
documents (texts, transparencies,
exercises, examinations) included,
available now and free (style free
software), then here are some useful
pointers.

[...] I have just updated a systems course
from the second cycle at CNAM, with
text, illustrations, exercises, examination
questions and answers. It is free and
downloadable. Usable according to the
practices of the free software (see
indications of copyright in on-line pdf
files).

[...] See:
http://deptinfo.cnam.fr/Enseignement/CycleProba
toire/SRI/Systemes (course using Ada).

Other CNAM systems courses, free and
on-line.

- First cycle:
http://deptinfo.cnam.fr/Enseignement/CycleA/A
MSI/

- Second cycle, including concurrency in
Ada: [see item (B) in next message. -- dc]

- Course on distributed systems:
http://deptinfo.cnam.fr/Enseignement/CycleSpeci
alisation/SAR

Claude Kaiser, Professeur, Titulaire de la
chaire D'Informatique Programmation,
Conservatoire National des Arts et
Métiers, Paris, France

From: Claude Kaiser <kaiser@cnam.fr>
Date: Mon, 12 Aug 2002
Subject: Re: [ada-france] taches
cc: ada-france@ada-france.org

Some info on courses available at CNAM
[about Ada tasking -- dc]

(A) Systems course with Ada tasking. In
the answers for the last examinations, I
give solutions with concurrent tasks and
protected objects [...]. In particular, see
the exam of June 2002. The course also
has a quick introduction to protected
objects, and an extensive bibliography.
[Cf. item 2) in previous message. -- dc]

(B) Course on concurrent programming.
Use of Ada for concurrency, hence
protected objects, tasks and rendez-vous,
etc. [...]
http://deptinfo.cnam.fr/Enseignement/CycleSpeci
alisation/ACCOV/

From: Jean-Pierre Rosen
*   <rosen@adalog.fr>*
Date: Thu, 8 Aug 2002
Subject: Re: [ada-france] taches
To: <ada-france@ada-france.org>

Don't forget the excellent book
"Programmation concurrente et temps réel
avec Ada 95" by Breguet and Zaffalon at
Presses Polytechniques et Universitaires
Romandes (in French).

[Cf. also "Two New References for Ada
95 (in French)" in AUJ 20-4 (January
2000), p.253. -- dc]

From: Zaffalon Luigi
*   <zaffalon@eig.unige.ch>*
Date: Mon, 19 Aug 2002
Subject: Re: [ada-france] taches
To: ada-france@ada-france.org

Concerning exercises & solutions [for the
above mentioned book -- dc].

Solutions as well as additional exercises
and their solutions are available at this
address:
http://eig.unige.ch/lii/PTR_En_Vrac.html

Luigi Zaffalon, Laboratoire
d'informatique industrielle, Ecole
d'ingénieurs de Genève, Suisse
http://eig.unige.ch/lii

Teaching Concurrent Programming

From: Daniel Feneuille <feneuile@univ-
*   aix.fr>*
Date: Fri, 16 Aug 2002
Organization: IUT Aix
Subject: Re: [ada-france] taches
To: ada-france@ada-france.org

[Extracts translated from French: -- dc]

*[...] before studying concurrent
programming in a given language, it is
really useful to have assimilated the*

*general concepts concerning parallel
computation and concurrent
programming.*

Well I do not agree, [...] one can perfectly
discover concurrent programming (with
Ada of course!) without extensive
preliminary study. This is what we do at
[the universities at Aix en Provence]. That
works very well and allows to go further
(with C++) and ... more quickly. By
always reserving Ada for an elite (i.e.
more "mature" students) one fuels the
legend that it is a difficult language. Since
1988 I claim the opposite and I prove it
each year. [...]

Plagiarism Detection

From: Alan Barnes
*   <barnesa@aston.ac.uk>*
Date: Wed, 17 Jul 2002
Organization: Computer Science, Aston
*   University*
Subject: Plagarism Detection
To: team-ada@acm.org

Can anyone recommend any software to
assist in plagiarism detection in Ada 95
programs? We have around 200 students
doing initial programming courses each
submitting 5 programming assignments (a
few hundred lines of code each) over the
course of 2 semesters. [...]

Dr. Alan Barnes, Computer Science,
Aston University, Birmingham, UK
http://www.cs.aston.ac.uk/~barnesa

From: Carlisle Martin C Dr USAFA/DFCS
*   <Martin.Carlisle@usafa.af.mil>*
Date: Wed, 24 Jul 2002
Subject: Re: Plagarism Detection
To: team-ada@acm.org

See, e.g.:
http://www.wired.com/news/topstories/0,1287,10
464,00.html

Martin C. Carlisle, PhD, Associate
Professor and Advisor in Charge,
Department of Computer Science, United
States Air Force Academy

URL:
http://www.wired.com/news/topstories/0,1287,10
464,00.html

[For the full text, see URL above. Some
extracts: -- dc]

Catching Computer Science Cheaters, by
Michael Stutz

Academic plagiarism has long been a
problem in computer science faculties, but
instructors and other university staff are
increasingly turning to a series of free,
Web-based tools to ferret out plagiarized
code and catch cheaters.

"Computers make plagiarism easier, but
also facilitate detection," said Michael
Wise, author of YAP, a Web-based
program that searches for "borrowed"
code in computer science homework
assignments. To deal with such
dishonesty, professors are using YAP and

other programs, including MOSS, or "measure of software similarity."

MOSS searches for similarities among programs written in the Ada, C, C++, Java, Pascal, Lisp, ML and Scheme programming languages. Professors submit batches of student programs to the MOSS server, then obtain the results minutes later via the tool's Web site, where a visual interface highlights suspect code in red.

[...] The MOSS algorithm is based on "code-sequence matching," says Alex Aiken, the program's developer. Aiken says MOSS does not analyze a program's algorithms - a task that is still too difficult. Rather, the program bases its findings on syntax, or the structure of the program itself. Aiken said that this method is more effective than counting the frequency of words in the program - the usual method of software plagiarism detection.

[...] "Computer science instructors have guessed that on any given assignment, between 5 and 20 percent of the students have collaborated 'beyond what is reasonable,'" said Kenneth C. Moyle, computing services coordinator for science faculty at McMaster University. Moyle's impression is that cheating is a serious problem in computer science courses because it is so easy to plagiarize a program by making small changes to alter its appearance.

[...] Aiken said that the cheating-detection programs stand a chance to reduce the incidence of plagiarism. "After students get used to the idea that there's a real risk of getting caught, then I think people will be more circumspect about cheating," Aiken said. [...]

From: Roger Racine
  <rracine@draper.com>
Date: Fri, 26 Jul 2002
Subject: Re: Plagiarism Detection
To: team-ada@acm.org

[...] I worry about any tool might come up with false positives.

[...] Please be certain before accusing. For a small assignment, given the specification, I would not be surprised to find 2 (or more) students with similar designs. I remember a conference talk on n-version software, where the author (the name escapes me) said that they needed to have (I think) 8 independent implementations before a significant number of design errors would be caught. Not only were the designs similar, they made the same mistakes! [...]

Roger Racine, Draper Laboratory, Cambridge, MA, USA

Rational's SEED Program

From: Inbrief Benelux (Rational Software)
  <inbriefbnl@Rational.Com>
Date: Fri, 30 Aug 2002

Subject: Rational Inbrief Benelux - August 2002
To: <inbrief-benelux@Rational.Com>

[...] Rational Software helps prepare students for software-based economy

Our enhanced Software Engineering for Educational Development (SEED) program currently reaches more than 20,000 computer science and software engineering students at more than 650 colleges and universities around the world.

The SEED program is a worldwide program through which Rational contributes its leading software and training materials to faculty and students at degree-granting institutions.

The program enables students to gain experience with Rational's software development tools, technologies and best practices while still in the classroom, helping decrease their learning curve when they start software development jobs after graduating.

[...] Read the press release:
http://www.rational.com/news/press/pr_view.jsp?ID=8329

Get more information on the SEED program:
http://www.rational.com/corpinfo/college_relations/seed/index.jsp [...]

Ada Training

[This information is included as examples of public Ada training courses: many are being organized regularly. For more, see also pointers in several previous AUJ issues. -- dc]

From: colbert@abssw.com (Ed Colbert)
Date: 29 May 2002
Subject: [Reminder] Public Ada 95 Class During Week of 10 June 2002 in Carlsbad CA
Newsgroups: comp.lang.ada

Absolute Software will be holding a public Ada 95 class on the week of 10-14 June 2002 in Carlsbad, CA. You can find a full description and registration form on our web site, 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.) [...]

From: John Robinson
  John@JohnRobinsonAndAssociates.com
Date: Thu, 6 Jun 2002
Organization: John Robinson And Associates Ltd
Subject: News Update - June 2002
To:
  News@JohnRobinsonAndAssociates.com

[...] Next Public Ada Course - Cheltenham - 1st-5th July

Following the success of our April course, we are returning to Cheltenham in July to run our next public Ada course. Once

again, this course will have two streams, one for Ada 83 and one for Ada 95.

The 5-day Ada course has just been through an upgrade, and is now better than ever. This upgrade is one of a series of new course releases which we will be announcing this year. [...]

URL:
  http://www.ddci.com/intro_to_ada95_class.shtml
Training: Introduction to Ada 95; September 5, 2002.
Type of Class: Open Enrolment

Are you looking to learn more about the capabilities, features and functionality of the Ada programming language? Find out what it has to offer you in developing complex software systems.

[...] This course will present the major new features in Ada 95 and how these features relate to Ada's support for data abstraction, information hiding, localization, and modularity. A short summary of the goals of Ada 95 will be reviewed as well as detailed descriptions on the key major enhancements in the areas of: Object-oriented programming, Real-time programming, Programming-in-the-large, Domain specific programming.

Designed for software engineers, managers, and quality assurance engineers interested in learning more about the Ada 95 language. Programming experience and familiarity with a high-level programming language are expected.

Ada -- The Most Trusted Name in Software!

Instructor: Dr. Joyce Tokar, has invested more than seventeen years of research and development in the improvement of embedded systems technology. A recognized leader in the Ada community, Dr. Tokar has received numerous awards for her contributions including the Outstanding Ada Community Contribution Award' 2000 ACM (Assoc. for Computing Machinery) SIGAda Conference.

[...] Location: DDC-I Phoenix Technology Center, Phoenix, AZ

Ada-related Tools

Booch Components

From: Simon Wright
  <simon@pushface.org>
Date: Sun, 2 Jun 2002
Subject: Booch Components 20020602
To: team-ada@acm.org

This release has been uploaded to http://www.pushface.org/components/bc/ and is mirrored at http://www.adapower.net/booch/

Features since 20020117:

Implementation changes: Improved closed Iterator (Visit, Modify) performance. Optimisations on BC.Containers.Trees.AVL.Validate, thanks to Steve Deller. Implemented "=" properly for Bounded and Dynamic Maps. Completed implementation of Maps.Are_Equal. Modifications to BC.Containers.Maps.Unbounded to allow the new Configuration_Demo to build with GNAT 3.14p (and GCC-3.1). It won't build with 3.13p, by the way, and has finalization problems with GNAT 3.15a.

New contributions: Anh Vo has contributed a pair of storage managers.

Other notes: The new storage pool management scheme, introduced in the 20011011 release, breaks GNAT 3.12. Converted the Tests and Demos page to report compiler compatibility.

[Cf. also same topic in AUJ 23-1 (March 2002), p.10. -- dc]

ASCL - Ada Standard Component Library

From: Michael Erdmann
*   <erdmann@snafu.de>*
Date: Thu, 04 Jul 2002
Subject: ASCL Prerelease 0.1.2
Newsgroups: comp.lang.ada

This is the first release of the Ada Standard Component Library (ASCL) project. The package may be downloaded via: http://sourceforge.net/projects/ascl

The idea of this library is to integrate already existing useful Ada 95 components into a library, which can be shared by the Ada community. Everybody in the community is invited to contribute or donate components to the ASCL project: http://ascl.sourceforge.net that have proved to be useful in the past. The primary objective of this project is to establish a reasonable code base and a first release in the 4th qtr of this year.

GCC 3.1 Released, Includes GNAT Sources

From: anh_vo@udlp.com (Anh_Vo)
Date: 15 May 2002
Subject: GNAT Sources Under GCC 3.1 Are Out
Newsgroups: comp.lang.ada

GNAT sources under GCC 3.1 are out today 15 May 2002 just in case nobody has noticed it. It was nicely bundled in a separate file. For convenience the pointer is: ftp://ftp.gnu.org/pub/gnu/gcc/gcc-3.1/

From: sjs@essex.ac.uk (Steve Sangwine)
Date: Thu, 16 May 2002
Subject: GCC 3.1 released
Newsgroups: comp.lang.ada

The GCC website reports release of GCC 3.1 on 15 May (yesterday). As has been noted in *comp.lang.ada* before, 3.1 includes the Ada 'front-end' as they call it, known to us as Gnat. [...]

[Status early Sep 2002, as seen on http://gcc.gnu.org/ : May 15, GCC 3.1 released; July 26, GCC 3.1.1 released; August 14, GCC 3.2 released. -- dc]

Status of GNAT in GCC 3.1

From: dewar@gnat.com (Robert Dewar)
Date: 16 May 2002
Subject: Re: GCC 3.1 released
Newsgroups: comp.lang.ada

What is the status of GNAT in gcc 3.1? Will it build on Windows and are there any outstanding issues?

It will build on windows and many other targets. There are known problems on all targets, but this version is in fairly good shape at least for fiddling around with!

From: Bobby D. Bryant
*   <bdbryant@mail.utexas.edu>*
Date: Fri, 17 May 2002
Subject: Re: GCC 3.1 released
Newsgroups: comp.lang.ada

FWIW, right after they forked the 3.1 branch I downloaded it and tried it out, and it seemed to work just fine on the various programs that I'm actively maintaining right now.

Bobby Bryant, Austin, Texas

From: Jeffrey Creem
*   <jeff@thecreems.com>*
Date: Sat, 18 May 2002
Subject: Re: GCC 3.1 released
Newsgroups: comp.lang.ada

But keep in mind that there _are_ problems, really.

Yup. There are problems. There are also problems in GNAT 3.14, 3.13, 3.12, 3.15 ... etc. I am not trying to minimize the issues in the 3.1 release (or say the other GNAT releases are bad!) however there are always problems in all releases. [...]

From: dewar@gnat.com (Robert Dewar)
Date: 18 May 2002
Subject: Re: GCC 3.1 released
Newsgroups: comp.lang.ada

But none of these releases failed ACATS tests and tests in our test suite. GNAT 5 [i.e. the GNAT version in GCC 3 -- dc] has significant numbers of failures in all categories on all targets. Several of these are of the form of incorrect code being generated silently and resulting in wrong results. I am not saying that the build is unusable, not at all, it will probably work fine for a lot of stuff, but it is still not at the product releasable level for us. Actually my current view is for our most up to date internal tree, which is quite a bit beyond the 3.1 release, so the 3.1 release may well have additional problems, we have not run the 3.1 release itself against our test suites at all. I

believe Laurent Guerby is working to make the ACATS suites usable in the FSF context.

One thing to understand here is that in the past when we have released a version like 3.15p it has passed all our internal qualification tests at the point corresponding to its internal freeze date. That's always been a criterion for any release (of GNAT Pro or the public version). A consequence was that the public versions were always quite a bit behind, but reasonably reliable.

Our new approach for public releases, integration into the GCC 3 tree, does not give any such guarantees. The plus is that it is far closer to our development wavefront, the minus is that it is likely to be somewhat unstable. Our feeling is that students can probably stay with 3.14p just fine, and hobbyists who want to fiddle with the latest and greatest and don't care whether it's 100% reliable can benefit from the GCC 3 version.

From: Jeffrey Creem
*   <jeff@thecreems.com>*
Date: Sat, 18 May 2002
Subject: Re: GCC 3.1 released
Newsgroups: comp.lang.ada

And of course I am not going to argue with Robert about the quality of any particular GNAT release since that would be somewhat silly.

But... The fact that past public releases pass all ACATS tests and 3.1 fails some and can produce incorrect code is somewhat bad however I strongly suspect that at some point in their life cycles each of those public releases produced incorrect code on some customers code somewhere.

Passing ACATS does give me a warm fuzzy but regression testing any new compiler release on my own code base is the only thing that really convinces me of the quality of any given compiler for my own use.

Still, I would not suggest that people use 3.1 for some important project with a real customer but then again I don't think it is really a great idea to use any of the other versions (of any complicated product) for that purpose without support either. [...]

P.S. I am sure I said something above that conflict with my first statement about not arguing with Robert about GNAT... I am glad I just said it was a silly idea and not a stupid one :)

[Robert Dewar responded: -- dc]

In fact everything you said makes perfect sense. No one is saying GNAT in 3.1 will fail on all programs, just that it does not meet our QA criteria yet and we know of too many programs on which it does fail.

Early Experiences with GNAT in GCC 3.1

From: Jeffrey D. Cherry
<yrrehcdj@hcetu.ten>
Date: Wed, 19 Jun 2002
Organization: Northrop Grumman
Subject: GCC 3.1 with GNAT ... Cool!
Newsgroups: comp.lang.ada

I would like to give my compliments to all who participated in incorporating GNAT into the GCC. It's a real treat to have one compiler installation, one command shell, and be able to build programs in Ada, C, C++, or FORTRAN 77 using the same shell and compiler. This is so cool!

A few weeks ago I downloaded the MinGW distribution of GCC 3.1 (dated 16 May 2002), installed it on my Windows 2000 machine, and compiled a bunch of rather simple legacy Ada, C, C++, and FORTRAN 77 programs. They all ran successfully. I installed the Win32 API and the Win32Ada bindings and compiled several more small Ada programs that used some Win32 services. These ran successfully as well. I then installed the Booch components and although there were several warnings during the compile step, all the tests and demo programs ran successfully. I compiled several of my Ada programs that utilize the Booch components and they ran successfully as well. I installed an old POSIX binding (Pascal Obry's Win32POSIX) and the tests in that distribution ran successfully. I had one Ada program that used the POSIX binding so I tried compiling it and it ran successfully as well.

Rather pleased with all this, I downloaded the GNU Pascal Compiler, installed it, and compiled a rather large suite of static code analysis tools (all written in Borland Pascal with Objects, v7). After some tweaking, they all compiled. I ran the resulting tools against some code I had analyzed a few years ago and checked the output against the archived copy. Except for the time and date of the analysis runs, the outputs were identical. It's been quite a productive month.

Granted these were rather trivial tests of the compiler and bindings, but the pessimist in me didn't think the first major distribution of the GCC with GNAT would be able to handle it. I, quite gladly, stand corrected.

Since I teach Ada, C, and C++ part-time at the local community college, this is a real blessing since all the students can now use the same compiler distribution. That may not sound like much, but you wouldn't believe how big a time-saver it is to have a single compiler configuration for all the classes.

[...] Once again, for anyone and everyone involved in incorporating GNAT into GCC, thank you, thank you, thank you!

Jeffrey D. Cherry, Senior IV&V Analyst, Northrop Grumman Information Technology

From: achrist@easystreet.com
Date: Wed, 19 Jun 2002
Subject: Re: GCC 3.1 with GNAT ... Cool!
Newsgroups: comp.lang.ada

Impressive indeed. Only about half of commercial software works that well out of the box. For so many pieces of software from diverse contributors to work together like that is something that should help promote Ada at least a little. [...]

GNAT and GCC 3.x for Mac OS X

From: jim hopper
<hopperj@macconnect.com>
Date: Fri, 19 Jul 2002
Subject: gnat on mac os x
To: GNAT Discussion List
<gnatlist@lyris.seas.gwu.edu>

BTW, how is GNAT based on GCC 3.x doing on MacOS X?

Its really doing very well. We are building full up Mac OS X Carbon apps with it. I don't remember when I saw a bug box last. We have the Apple IDE working really well with Ada. It's one of the nicest development environments for ada I have worked with on any platform.

[And from another message: -- dc]

www.adapower.net/macos

Things are very nice now, will be MUCH better in a month when the next big release of MacOS X is to be released.

[See also "GNAT for Mac OS X/Darwin" in AUJ 23-1 (March 2002), p.10. -- dc]

Fuzzy Sets for Ada

From: Dmitry A. Kazakov
<mailbox@dmitry-kazakov.de>
Date: 28 May 2002
Organization: Computer Science, University of Melbourne
Subject: [ANNOUNCE] Fuzzy sets for Ada 3.0
Newsgroups: comp.lang.ada,comp.ai.fuzzy

I have uploaded the version 3.0:
http://www.dmitry-kazakov.de/ada/fuzzy.htm

Now it supports fuzzy boolean logic as well as fuzzy integer and float numbers (built upon intervals).

Matrix Manipulation Code

From: Craig Carey
<research@ada95.ijs.com>
Date: Sat, 08 Jun 2002
Organization: Customer of Mercury Telecommunications Ltd

Subject: Re: Making a package safe for use by multiple tasks
Newsgroups: comp.lang.ada

Some links to Ada matrix 95 source code doing matrix multiplication.
Ada95 matrix package (54KB tar.gz file), Drexel Fusion Laboratory:
http://dflwww.ece.drexel.edu/research/ada/

Some vector operations in PragmAda (which is a thing going into the new ASCL source code library,
http://ascl.sourceforge.net/):
http://home.earthlink.net/~jrcarter010/pragmarc.htm

Some matrix pivoting source code to solve linear equations:
http://lglwww.epfl.ch//Team/MW/mw_components.html

Some Ada 95 maths software, simulated annealing: http://www.taygeta.com/ada.html , /annealing/simanneal.html

Multiply 2 matrices:
http://www.almink.com/school/matrix.html

Bindings to a FORTRAN library:
http://topo.math.u-psud.fr/~sands/Programs/BLAS/

Zeller's Algorithm - Day-of-Week for Dates in Gregorian Calendar

From: Thomas Wolf
<t_wolf@angelfire.com>
Date: Wed, 24 Jul 2002
Subject: Re: Zeller's Algorithm
Newsgroups: comp.lang.ada

I am trying to find a Ada implementation of Zeller's algorithm that works! I have found several on the web but none of them have worked right. [...] I have had similar weird results using other similar Ada Zeller versions I've found on the web.

You might try the Util.Calendar packages from my Util subsystem, available at the URL
http://home.tiscalinet.ch/t_wolf/tw/ada95/util/

Util.Calendars.Universal.Day_Of_Week produces the correct results. See the test driver in ./test/calendar_test.adb for some examples. The Util subsystem is published under the GMGPL.

From: Thomas Wolf
<t_wolf@angelfire.com>
Date: Tue, 30 Jul 2002
Subject: Re: Zeller's Algorithm
Newsgroups: comp.lang.ada

What is the reference for this algorithm. I learned it as a child (it is something you can learn to compute in your head quite easily).

Chr. Zeller, "Kalender-Formeln", Acta Mathematica, Vol. 9, pp. 131-136. (Nov. 1886). In German.

See also the URL:
http://www.merlyn.demon.co.uk/zeller-c.htm an excellent page on Zeller's congruence.

From: t_wolf@angelfire.com (Thomas
* Wolf)*
Date: 31 Jul 2002
Subject: Re: Zeller's Algorithm
Newsgroups: comp.lang.ada

> *While reviewing the latest Risks*
Digest, I came across the following item
titled "Possible day-of-week error"
concerning the Zeller algorithm:
http://catless.ncl.ac.uk/Risks/22.18.html#subj10.1

That message in the Risks Digest relates to the possibility of the left operand of the final "mod" operation in Zeller's formula becoming negative in some cases. This may be a problem in languages such as C, where e.g. -30 % 7 = -2. In Ada 95, using "mod", this is not a problem, for A mod B is defined to return a modulus in the range 0 .. B-1 for positive B, regardless of the sign of A. If you want to avoid a negative left operand altogether, use the formula:

Z := (Integer (Day) + (13 * M - 1) / 5 + Y + Y / 4 + Century / 4 + 5 * Century) mod 7;

instead of Zeller's :

Z := (Integer (Day) + (13 * M - 1) / 5 + Y + Y / 4 + Century / 4 - 2 * Century) mod 7;

Adding 7 * Century doesn't change the final result (because it's a multiple of 7), but ensures that the left operand is always positive.

From: Hyman Rosen <hyrosen@mail.com>
Date: Thu, 01 Aug 2002
Subject: Re: Zeller's Algorithm
Newsgroups: comp.lang.ada

I found an interesting paper that seems to be more than you ever wanted to know about computer integer division.
http://www.cs.uu.nl/~daan/papers/divmodnote-letter.pdf

PNG_IO - I/O of Portable Network Graphics Files

From: Jacob Sparre Andersen
* <sparre@nbi.dk>*
Date: Fri, 21 Jun 2002
Organization: Centre for Chaos and
* Turbulence Studies, Niels Bohr Institute*
Subject: Re: Ada + Web + CGI
Newsgroups: comp.lang.ada

[...] You could use PNG_IO to create PNG images with the data:
http://privatewww.essex.ac.uk/~sjs/png_io/png_io.html [...]

[From that page: -- dc]

PNG_IO is an Ada 95 Portable Network Graphics (PNG) coder/decoder. It is designed to provide Ada 95 programmers with direct access to images and graphics stored in PNG format files. PNG is an image file format supporting greyscale and colour images with and without alpha channels. It is patent-free and offers good levels of lossless compression.

Graphics in Ada

From: Pascal Obry <p.obry@wanadoo.fr>
Date: 05 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

I want to have a go programming a
game for Ada, except I don't know how
to use graphics, [...] Does anyone know
where to get tutorials for graphics ?

[Cf. also "Graphics Packages" in AUJ 23-1 (March 2002), pp.12-13. -- dc]

What OS? What kind: 2D or 3D game? If the later and you want to run on Windows and UNIX the way to go is OpenGL. Close to Direct3D and there is very good drivers for the 3D video cards. You could have a look at:
http://www.libsdl.org/index.php

SDL claims to be a cross-platform, quick, used by some nice software and there is an Ada binding. I have never tried it though...

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

http://perso.wanadoo.fr/pascal.obry

From: David Marceau
* <davidmarceau@sympatico.ca>*
Date: Mon, 05 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

1) Platform independent Gtk. GtkAda binding: http://libre.act-europe.fr/GtkAda/
This one I tried and it has a lot of components to offer. Check out the demos very carefully there are a lot of hidden perls here. Tree controls, tables, interconnecting node graphs usable for creating UML tools. It is most likely you would want to use this since it runs on Linux/Unix and windows.

2) Platform independent SDL. AdaSDL binding:
http://sourceforge.net/projects/adasdl
I haven't tried this one.

3) Linux 2d/3d graphics with Ada.
http://www.mysunrise.ch/users/gdm/e3d/engine3d.tgz

You'll need svgalib:
http://www.arava.co.il/matan/svgalib/
This is straight console graphics. There are no XWindow or Windows in this one. Potentially embeddable. What do I mean potentially? I mean you could if you wanted to :)

4) Proprietary Windows95/NT/2000...
If you want to do game programming in Ada95 with just MS-Windows in mind, then consider using gnatcom [Cf.
http://www.adapower.com/ -- dc] in order to reuse direct2d, direct3d, directshow filters.

5) Consider using jgnat/cafe 1815 to connect to JAZZ (an open-source Java zoomeable user interface built on top of swing/awt).
http://www.cs.umd.edu/hcil/jazz/download/index.shtml
It's less likely that you use this however the design and JAZZ's patterns are inspiring source code to rewrite in Ada/svgalib/sdlada/gtkada.

Again there are many hidden perls in the jazz source code, architecture and design patterns. It's worth the look.

From: David Botton <David@Botton.com>
Date: Mon, 5 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

If you are using Windows, you may want to take a look at GWindows and go through the tutorials
http://www.adapower.com/gwindows [...]

AdaSDL - Binding to Simple DirectMedia Layer (SDL)

From: Maarten Woxberg
* <marwo264@student.liu.se>*
Date: Sun, 11 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

António Vargas wrote:
> *I'm the creator of the AdaSDL Ada*
binding. [...] I will continue to improve
AdaSDL if I get some feedback. I wish
to learn and get some constructive
criticism from others. My actual e-mail
is: antoniovargas@oninet.pt

[Cf. also same topic in AUJ 22-3 (September 2001), p.141. -- dc]

I would really like to get the latest version of your AdaSDL. I'm in the beginning of development of a game in Ada and would greatly appreciate a binding that is mostly compatible with SDL 1.2. Sound is not required. [...]

From: Eric Merritt
* <cyberlync@yahoo.com>*
Date: Mon, 12 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

Just as a heads up. I am using AdaSDL for a game client I am working on. So far it's working pretty darn well. I do have some suggestions if you are interested in them.

[...] I most definitely appreciate the work you put into on this. It's making my life a whole lot easier. The fact that AdaSDL exists is what gave me the option of choosing Ada for the client, otherwise it would have ended up being Objective-C or something similar.

From: antoniovargas@oninet.pt (António
* Vargas)*
Date: 13 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

It's good to know that it works well for others to. Since my last post I have received several E-mails from people interested in the project. So, I will make a new release with some small corrections and additions (such us the use of truetype fonts over SDL/OpenGL) and a better intallation process/documentation for Linux/Windows.

[...] It is a thin binding. A thick binding would be a good add on in order to do Ada code in Ada style (not Ada code influenced by the binded underlying C code). I've started something but is far from complete.

The main advantage of the present thin binding is related to the fact that it is very easy to port C/SDL software. You might have notice that in the several examples of the package. [...]

Antonio M. F. Vargas, Barcelos, Portugal

VAD - Visual Ada Developer

From: Stas <dulman@attglobal.net>
Date: Mon, 3 Jun 2002
Subject: Announce Visual Ada Developer
  (VAD) version 6.0
Newsgroups: comp.lang.ada

VAD (Visual Ada Developer) is a Tcl/Tk oriented Ada-95 (TCL) GUI builder portable to different platforms, such as Windows NT/9x, Unix (Linux), Mac and OS/2. You may use it as IDE for any Ada-95 (C, C++, TCL) project. You may use it to build Ada sources and (or) TCL scripts.

Used software: GNAT 3.14p: Ada-95 compiler: ftp://cs.nyu.edu/pub/gnat

TASH 8.02: by Terry J. Westley http://tash.calspan.com/

RAPID-1: by Martin Carlisle [ftp://ftp.usafa.af.mil/pub/dfcs/carlisle/usafa/rapid/index.html -- dc]

TCL/TK 8.0.5 & 8.2.3: http://www.scriptics.com/software

TCL/TK 8.3.4.2 & 8.4.0.b2: http://tcl.activestate.com/software/tcltk/

VAD 6.0 has four realizations: for tcl/tk 8.0.x, 8.2.3, 8.3.4.2 and 8.4.0.b2 (last version). You need to install and to check tcl/tk before using VAD. From version tcl/tk 8.4.0.b2 on the ActiveState distribution includes many of VAD used packages (Itcl, Img, Tktable, BWidgets, Tkhtml and so on). You may choose the needed version at link time. (I recomend to work with tcl/tk 8.3 or 8.4). VAD binaries are compiled for tcl/tk 8.4.

[Large number of supported packages removed: see web-page for full announcement. -- dc]

VAD 6.0 is available in http://www.websamba.com/ldulman/vad.htm You may dowload sources [...] and binaries [...] (Windows 9x/NT) [...] (OpenLinux 2.x, RedHat 6.x, 7.x)

Leonid Dulman (dulman@attglobal.net)

[Cf. also AUJ 22-3 (September 2001), p.142, for more details and pointers. -- dc]

AdaLDAP - Binding to OpenLDAP Library

From: Jeremy Cowgar
  <develop@cowgar.com>
Date: Thu, 06 Jun 2002
Subject: Announcement: Ada LDAP Binding
Newsgroups: comp.lang.ada

As previously discussed I stated that I would place into open source the LDAP binding that I am using for one of my projects. I have created an account on savannah.gnu.org for this.

The binding in it's current state should only be used to see that it works. Only an incomplete thin binding exist that is difficult (same as C) to work with. The thin binding will go through a few changes before it will be considered a "firm" API. Once this is accomplished, a thick Ada binding will be created to make LDAP much easier to use.

The current download contains the incomplete (but functional) thin binding and a demo program that queries the University of Akron's LDAP server for all entry's whose name is John* ...

Take a peek at: http://savannah.gnu.org/projects/adaldap/ for all the details. If anyone would like to contribute to the project, it will be appreciated.

TAP - Thick Ada-Prolog bindings

From: Alexandre E. Kopilovitch
  <aek@vib.usr.pu.ru>
Date: Tue, 20 Aug 2002
Subject: ANNOUNCE: TAP (Thick Ada-
  Prolog) bindings
Newsgroups: comp.lang.ada

I just made available my TAP (Thick Ada-Prolog) bindings at http://www.tarkvara.com/tap

These bindings assume GNAT compiler (3.13p or 3.14p) for Ada side, and Amzi Logic Server (see: http://www.amzi.com) (version 6.2) for Prolog side.

All command files in the distribution are for Windows (and the whole thing was tested under Windows 2000 only). Hopefully these bindings may be used with Unix/Linux also (both GNAT compiler and Amzi Logic Server are available for Solaris and Linux), but I never tried that.

This is release 0.1 of the bindings, [...] You will find in the distribution the full source code of the bindings, the manual and tutorial (in HTML form), a complete example (which apparently works), and some other stuff, including the text of the LGPL license.

Send your critique, error reports, suggestions, condemnations and praises, relevant to these bindings, to aek@acm.org or aek@vib.usr.pu.ru

Alexander Kopilovitch, Saint-Petersburg, Russia

[Cf. also "Interfacing to Prolog" in AUJ 23-2 (June 2002), p.74, and "Prolog in Ada" in AUJ 20-4 (January 2000), p.236. -- dc]

GNADE - GNAT Ada 95 Database Development Environment

From: Michael Erdmann
  <erdmann@snafu.de>
Date: Sat, 08 Jun 2002
Subject: Release of GNAT Database
  Development Env. 1.3.5
Newsgroups: comp.lang.ada

The source code release 1.3.5 of the GNADE project, which replaces the previous version 1.3.4b is available at: http://gnade.sourceforge.net/ http://sourceforge.net/projects/gnade for the Linux, Windows and Solaris 2.8 platform. This environment allows, using the GNAT Ada 95 compiler, to implement Ada 95 applications invoking relational database products such as MySQL, PostgreSQL, MimerSQL and Oracle.

[Cf. also "On Databases and Ada" & "AdaIC Press Release - Ada Expands into Linux Market" in AUJ 23-1 (March 2002), p.14 & p.28, and "GNADE - GNU Ada Database Environment" in AUJ 22-3 (September 2001), p.144. -- dc]

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

[On Fri, 05 Jul 2002, version 1.3.6a was released. -- dc]

APQ - PostgreSQL Ada95 Binding

From: Warren W. Gay VE3WWG
  <ve3wwg@cogeco.ca>
Date: Sat, 03 Aug 2002
Subject: PostgreSQL Ada95 Binding APQ
  1.0 Released
Newsgroups: comp.lang.ada

The APQ 1.0 Ada95 binding for PostgreSQL is a very easy to use binding for SQL interactions with the database. This binding uses the C language libpq library and does NOT require ODBC software. This makes PostgreSQL available to Ada95 programmers with the minimum of software prerequesites.

The following features are provided: it is a thick binding (for a natural Ada95 experience); it is very simple to use: only 3 objects to know (Connection_Type, Query_Type, Blob_Type); support for strong Ada95 data types; generic function and procedure support; full NULL indicator support; full BLOB support; Ada95 stream I/O for blobs; Additional DATE, TIME and TIMESTAMP support routines; ACL License (Ada Community License); Experimental Decimal child package support; 100+ page reference manual (almost every function has an example); no embedded SQL precompiler needed.

The following were APQ design goals: very simple to use (#1 priority); easy to read both SQL and Ada code; reliable (exceptions for error conditions); strong blob support; no C language interfaces or types ☺ easy to install (no ODBC! Requires only Postgres libpq); easy to debug (To_String on Query_Type for example returns the full SQL text used).

Tested and developed on FreeBSD 4.4, using GNAT 3.13p. While untested for Linux, it should install OK on modern Linux platforms. It should port well to other UNIX platforms as well. [...] You can download the source+manual, or just the PDF manual alone from http://home.cogeco.ca/~ve3wwg

[One day later: -- dc]

I have re-released APQ as version 1.1, with changes that licenses it under a dual-license as Florian suggested. The user/distributor can choose either the ACL [Ada Community License -- dc] or GPL license. Apparently the ACL can be somewhat restrictive for the distributor of the source code.

Please find the changes in the manual and source code, found at: http://home.cogeco.ca/~ve3wwg

From: Warren W. Gay VE3WWG
* <ve3wwg@cogeco.ca>*
Date: Fri, 09 Aug 2002
Subject: Release APQ 1.2 Solves Blob
* Stream I/O Performance Problem*
Newsgroups: comp.lang.ada

Sorry for the frequent software updates, but the stream I/O performance issue for PostgreSQL blobs, formerly made the stream interface virtually unusable. This problem has been corrected in the APQ version 1.2 release, and there is no longer any reason to avoid using blobs.

Download the manual and/or the Ada95 binding at: http://home.cogeco.ca/~ve3wwg/

What's new in APQ 1.2?

Buffered stream I/O for blobs. Unbuffered stream I/O is still user selectable. Blob_Flush procedure added. Blob_Create and Blob_Open have new optional Buf_Size argument. Fixed error recovery in Blob_Create (it now releases a successfully created, but not opened

new blob - this happens when Blob_Create is attempted outside of a PostgreSQL transaction). Added End_of_Blob function for convenient sequential processing of a blob. Fixed some formatting problems in the code examples of the manual. Expanded the troubleshooting chapter in the manual.

Tetris Game

From: Preben Randhol
* <randhol+ada@pvv.org>*
Date: Fri, 5 Jul 2002
Subject: Re: read from stdin
Newsgroups: comp.lang.ada

Check: http://www.tc.umn.edu/~puk/tetris.txt

[From the author: -- dc] "Ada source code for console-mode Tetris game for testing Ada 95 and Ada 83 tasking. Works for GNAT (Solaris, Win, DOS) and Ada 83 OpenAda compilers."

Preben Randhol, http://www.pvv.org/~randhol

«For me, Ada95 puts back the joy in programming.»

Ada for Game Programming

From: tmoran@acm.org
Date: Sun, 04 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

I know Ada isn't the best language for games, [...]

You know wrong. It can do drawing as well as anything, plus it has built in multi-tasking and timing and even multi-computer distributed programming.

From: David Botton <David@Botton.com>
Date: Mon, 5 Aug 2002
Subject: Re: Graphics with Ada ?
Newsgroups: comp.lang.ada

[...] BTW, Ada is a VERY_ good language for game programming! I wouldn't be surprised if a few aircraft simulators were written in it ☺

From: Caffeine Junky
* <nospam@hotmail.com>*
Date: Mon, 05 Aug 2002
Newsgroups: comp.lang.ada
Subject: Re: Graphics with Ada ?

I know Ada isn't the best language for games, [...]

I use to think this also. However my understanding and skills with the Ada language have mushroomed recently (hence my lack of posting questions to comp.lang.ada) and as a result I have discovered that Ada is in fact an excellent tool for programming games. I'm finding in particular that Ada is very well suited for doing simulations and games where "game physics" is a significant concern.

Generally I leave the details of actually drawing the graphics to the screen up to external libraries, mostly because my skills in that particular area are rather

weak at the moment. I do everything else in Ada95 though. (Such as maintaining a player database, doing the various calculations and tests, memory management, etc...).

However, due to Ada95's "streamlined" use of Boolean, Machine, and Storage_Pool types, I don't see graphics as being a particularly difficult hurdle. After all, it's often used to create flight simulators.

This is just an opinion. I have never used Ada in a professional context, so I can't really claim to have an authoritative opinion. However from a hobbyist standpoint, it's very effective.

Documentation Tools for Ada Sources

From: Michael Erdmann
* <Michael.Erdmann@snafu.de>*
Date: Sat, 11 May 2002
Subject: Q: Generating Documentation
* from Ada Sources?*
Newsgroups: comp.lang.ada

I am looking for open source tools which are generating documentation out of package specifications. Such a tool should not just list the items in the specification but also extract the comments belonging to these items into a reasonable format. Who knows more about the topic?

[And in response to: -- dc]

For a well written Ada spec, it would seem that the Unix tool cat would do what you want, or possibly something like gnathtml. I think you need to be a whole lot more specific about what you are looking for. What else is there in a spec except items and comments?

The point is that Ada spec. come along with comments which explains certain aspects of an interface. I like to generate from the *.ads file a docbook (SGML) formated manual page which contains the explanatory text. This mean for example the package name goes into the section name, the description into a paragraph, the spec. itself goes into a program listing section.

From: Randy Brukardt
* <randy@rrsoftware.com>*
Date: Sat, 11 May 2002
Subject: Re: Q: Generating Documentation
* from Ada Sources?*
Newsgroups: comp.lang.ada

We developed such a tool for creating the Claw documents (it outputs HTML and .RTF for WinHelp), but it's not open source and its rather tied to our Claw coding conventions. What it does is preprocess the specifications into an intermediate file, which can be updated manually to correct mistakes, add links and formatting, and the like. There is also some tools for merging files (so that new versions of packages don't lose the hand-

work from previous versions). I thought about making a GUI for it, and making it available as a tool, but I didn't think there was likely to be enough demand to make it worthwhile. If you don't find something else, drop me a line and we can discuss making it available to you.

[Tom Moran <tmoran@acm.org> added: -- dc]

The tool for Claw only requires the public part of specifications to be parsable with an ayacc Ada grammar. The only semantic requirement I recall is that things are assumed to be defined before use. It would be nice if there was a cut down, highly forgiving, version of ASIS to handle incomplete programs or situations where a compiler bug prevents compilation and thus prevents use of the current ASIS.

[And on conventions in coding style, so the tool "knows" which comments are "relevant": -- dc]

We went a different way, since most of the source files already existed, including customer's copies, and we didn't want to change them. We have a database of information, for instance every entity has a Remarks section (which of course may be empty) and a Spec section. Subprograms additionally have an Exceptions Raised section, and so forth. These are filled in as much as possible by a program that scans source files for specs, links, comments saying "Raises ..." directly following a subprogram etc. But this is viewed as just a convenience. We expect most of the text to be entered, or at least modified by, a human writer. Syntactic information like the file something is declared in, the package to "with" for it, any predefined constants of a type, etc comes from scanning the source code and is not modified by the documentation writer.

From: Ira D. Baxter
* <idbaxter@semdesigns.com>*
Date: Sun, 12 May 2002
Subject: Re: Q: Generating Documentation
* from Ada Sources?*
Newsgroups: comp.lang.ada

The DMS Software Reengineering Toolkit can parse Ada, build ASTs, and then use custom code to extract what you want. (It is kind of like ASIS, for a whole set of languages; not as complete as ASIS wrt Ada, but more capable than ASIS in the DMS can actually change the source tree and then prettyprint the changed result. The latter capability isn't needed for documentation extraction).

Part of the customization process can control how many syntax faults are acceptable before it gives up (at the price of getting a syntax tree with error insertions that might not make sense, but error recovery is a tough topic).

For the original poster, who wants to extract certain information and then reformat the way he likes, I'd suggest extracting the information he needs and dumping as XML text. One can use XSLT to reshape the result pretty much as you like. This decouples extraction from formatting. See:

http://www.semdesigns.com/Products/DMS/DMS Toolkit.html

Ira D. Baxter, Ph.D., CTO Semantic Designs, Inc.

From: Stephen Leake
* <stephen.a.leake.1@gsfc.nasa.gov>*
Date: 13 May 2002
Organization: NASA Goddard Space Flight
* Center (skates.gsfc.nasa.gov)*
Subject: Re: Q: Generating Documentation
* from Ada Sources?*
Newsgroups: comp.lang.ada

Have you tried AdaBrowse? [Cf. also
"AdaBrowse - A Javadoc for Ada95"
below. -- dc]

Michael Erdmann writes:
Thanks I have tried it. What I am
interested is more centered on
processing text in the source.
[AdaBrowse] does not care about the
contents of the comments. It is left to
the user to build in tags or not.

AdaBrowse is Open Source, so you can enhance it. Surely an ASIS-based tool is the right way to go for an Ada documentation project :). At least, if you need to go beyond cat (I agree with Robert Dewar here; just write good comments!).

What I am looking for is a tool which
maps the relevant part of the comments
in a package spec. and transforms this
into a docbook document.

This should be easy to add to AdaBrowse. You do have to establish conventions in coding style, so AdaBrowse+ can know which comments are "relevant". You want a somewhat elaborate syntax in the comments. AdaBrowse at least gets you the top level source traversal, and lets you write the comment parsing in Ada. Use GNAT.Spitbol or GNAT.Regexp, or Ada.Strings.Fixed, or OpenToken.

Any way, AdaBrowse is based on ASIS,
which requires that the module is
compilable which I cannot guarantee.

Why not? Surely you want to know that the document is accurate, which means that the Ada code compiles! Otherwise you could have inconsistent information! [...]

From: srevets@geol.uwa.edu.au (Stefan
* Revets)*
Date: Tue, 14 May 2002
Organization: University of Western
* Australia*
Subject: Re: Q: Generating Documentation
* from Ada Sources?*
Newsgroups: comp.lang.ada

Literate programming is another option. There is an AdaTangle and AdaWeave around, implementing the ideas of Donald Knuth.

[And in response to: -- dc]
It is Ada 83 though.
ftp://www.ctan.org/pub/tex/web/ada/aweb/

True. But Norman Ramsey wrote Spidery Web which allows one to generate *Tangle and *Weave for about any language desired. Using awk, you define the things you desire to be done with the language of choice, process it and build both tangle and weave from it. Norman Ramsey included an Ada.spider file: it shouldn't require that much investment of time to get an Ada version.

[ftp://www.ctan.org/pub/tex/web/spiderweb/ -- dc]

Stefan A. Revets, Department of Geology & Geophysics, University of Western Australia Nedlands, Australia

From: erreur <rogspr@newdeal.ch>
Newsgroups: comp.lang.ada
Subject: Re: Auto-Documenters for Ada
Date: Thu, 30 May 2002

I've got a ton of these tools for
C/C++/Java, but have yet to find one for
Ada that is highly customizable (i.e.
allows me to enter my own choices for
tags, can give HTML outputs, etc...).
Can someone in here recommend a
decent one? [...]

Try AdaDoc (free under GPL licence)! Its goal is to create a documentation in different format (currently only html is available) from a specification package.

[Cf. also "AdaDoc - Html Generator for Ada Package Specifications" in AUJ 22-4 (December 2001), pp.200-201. -- dc]

Release 2 of AdaDoc would be available in July 2002. It will support these points and add the following feature: different format output (html, LaTeX, docBook). It would be easy to anybody who knows Ada95 or C to add a format output (from the XML temporary file).

From: Deborah Torrekens
* <deborah@phidani.be>*
Date: Tue, 4 Jun 2002
Subject: Re: Auto-Documenters for Ada
Newsgroups: comp.lang.ada

You might want to take a look at the RainCode Roadmap
http://www.raincode.com/AdaRoadmap.htm
It generates html pages with the whole source code, annotated by hyperlinks on every variable, function, package, etc., cross-references and a number of metrics. On the website, you'll find an example of the roadmap on the GNAT sources.

AdaBrowse - A Javadoc for Ada 95

From: Thomas Wolf
* <t_wolf@angelfire.com>*

Date: Tue, 2 Jul 2002
Subject: ANN: AdaBrowse 2.1
Newsgroups: comp.lang.ada

I've published version 2.1 of AdaBrowse.
It's available at the URL
http://home.tiscalinet.ch/t_wolf/tw/ada95/adabro
wse/
in source form and as a pre-built
executable for Win NT/2k and GNAT
3.14p.

[Cf. also same topic in AUJ 23-1 (March
2002), p.19. -- dc]

Changes from the previous version (2.01):

* Start tags of user-defined HTML
entities may now have attributes, which
can be referenced in the replacement.

* AdaBrowse 2.1 supports user-defined
formatting rules for descriptions, giving
lots of flexibility in particular for
generating HTML from legacy code.

* Some corrections in HTML generation:
no more misplaced <P> tags, typo
correction (2.01 generated a MTEA tag
instead of META), improved handling of
unknown tags, sxtray end tags, and of
character entities, new "<!DOCTYPE"
string.

* The makefile now tries to figure out the
name of the compiler called by gnatmake,
and sets up AdaBrowse to use that as the
default name. [...] note: in any case, you
can still override the default by giving an
explicit compile command in a
configuration file.

AdaBrowse 2.1 has passed all my
regression tests; the generated HTML has
been verified to be compliant to the
"HTML 4.01 Transient" DTD. See the
included user's guide on how and where
to submit bug reports.

On a side-note: the Util subsystem (a
collection of generally useful Ada 95
packages, published under the GMGPL)
used by AdaBrowse and available at the
URL
http://home.tiscalinet.ch/t_wolf/tw/ada95/util/
also has been updated. It includes now
many more useful utility packages, such
as a thick (stream-based) binding to the
popen/pclose routines available on Win32
and Linux, extended calendar support,
time formatting operations, utility string
operations, operations on path names, a
configuration file reader, environment
variable substitution, and much, much
more...

The full AdaBrowse-generated reference
for these packages also is online at
http://home.tiscalinet.ch/t_wolf/tw/ada95/util/doc
This documentation is also included in the
tarball.

From: Thomas Wolf
  <t_wolf@angelfire.com>
Date: Wed, 3 Jul 2002
Subject: Re: AdaBrowse 2.1
Newsgroups: comp.lang.ada

David@Botton.com wrote:
I've made available the GNAT-3.14p
Runtime on AdaPower at
http://www.adapower.com/lang/gnatrun/packages
.html
using AdaBrowse 2.01 (I didn't have a
chance to update yet to the new
version).

Nice. I also noted that the GWindows
reference has been generated with
AdaBrowse. Hm. I somehow didn't think
that anybody would like to process the
standard Ada library with AdaBrowse [..]
After having looked at the generated
docu, I decided to add a -g command-line
option to AdaBrowse that makes it
generate cross-references to items from
the standard library, too. [...] Hence, the
current version of AdaBrowse is 2.11,
available now at the URL
http://home.tiscalinet.ch/t_wolf/tw/ada95/adabro
wse/

AdaBrowse 2.11 has passed all my
regression tests (which now include
generating a full docu for the standard lib
of GNAT 3.14p).

New features:

1. A -g command-line switch. If set,
AdaBrowse generates cross-references to
items from the standard library (except
those from package Standard), too.

2. A new configuration file key
"Refs_To_Standard"; its value must be
"True" or "False" (without the quotes). If
True, same as "-g" on the command line.
If False, switches off generation of cross-
refs to standard items.

3. Bug correction: 2.1 had a bug which
made it handle "Path" prefixes case-
sensitively. Would work only if the unit
name prefix in the key was given in all
lowercase. I.e. "Path.Ada = ..." wouldn't
work, but "Path.ada = ..." worked. This is
corrected in 2.11; both work now.

4. Major speed improvement: AdaBrowse
tries to minimize the number of times
ASIS contexts are opened and closed.
This appears to be a time-consuming
operation in ASIS-for-GNAT, and also
seems to incur memory leaks.

BTW, I tried generating the HTML docu
for GNAT's standard lib [...]. I first
generated all the *.adt files, and then let
adabrowse run on all of them. Before the
speed optimization (#4 above), that took
92 minutes and AdaBrowse used up
180Mb of virtual memory on my
machine. After the optimization, this now
takes about 1 minute and uses some 5Mb
of memory.

Macks - Safe Physical Units Handling in Ada

From: Fraser Wilson
  <newsfraser@blancolioni.org>
Date: 21 May 2002
Subject: Safe Units Handling in Ada

Newsgroups: comp.lang.ada

After a recent discussion, I put together
Macks, a small program that generates
Ada packages for dealing with physical
units in a type safe way.

You can now download Version 0.0.1
(that's confidence) from [see further --
dc]. Instructions are in the README, but
if you have GNAT, then all you have to
do is this: gnatmake -o macks macks-
driver [...]

[In response to "What is the derivation of
the word "Macks"?": -- dc]

(Meters/Moles), Amperes, Candela,
(Kilograms/Kelvin), Seconds. I probably
spent too much time thinking about that.
:-)

[One day later: -- dc]

Ah, the best-laid plans. The 'or' feature
doesn't work, and the logic was sloppy.
So here's a new version [0.0.3] which
fixes that. [...] I've also added a file called
macks.grammar, which specifies the
grammar of the .macks file. [...]

From: Fraser Wilson <fwilson@anago.nl>
Date: Thu, 23 May 2002
Subject: Re: Safe Units Handling in Ada
Newsgroups: comp.lang.ada

Perhaps you could explain how it works
and give a simple example.

Yes, that's a good idea. The important bits
as far as units are concerned is under the
Macks package hierarchy. Macks.Parser
reads a source file which describes the
system, and Macks.Writer emits Ada
source code. Macks.Table is a simple
symbol table.

Each unit type in the Macks source file
has a number of operator functions
generated for it. Common to all units are
the following:

function "*" (Left, Right : Unit) return
Unit is abstract;
function "/" (Left, Right : Unit) return
Unit is abstract;
function "*" (Left : Float_Type; Right :
Unit) return Unit;
function "*" (Left : Unit; Right :
Float_Type) return Unit;
function "/" (Left : Unit; Right :
Float_Type) return Unit;
function "/" (Left : Unit; Right : Unit)
return Float_Type;

Derived units have in addition operators
corresponding to their definition or
definitions in the .macks file; for
example, if "Speed is Meter / Second;"
then the following operators are
generated:

function "/" (Left : Meter; Right : Second)
return Speed;
function "*" (Left : Speed; Right :
Second) return Meter;
function "*" (Left : Second; Right :
Speed) return Meter;

function "/" (Left : Meter; Right : Speed) return Second;

The file example.macks contains a simple Macks specification, which only deals with a few derived units. However, that's enough to create an 800 line Ada package spec (which is what motivated this sort of thing in the first place).

Macks.Driver is the main procedure; with Gnat you can use the command "gnatmake -o macks macks-driver.adb" to get an executable; this can be invoked with "macks example.macks" for example. <filename>.macks generates <filename>.ads and <filename>.adb. To use the generated files, with and use/use type them, then declare your quantities and act natural. If operations that should work don't, let me know.

I hope this helps; more thorough documentation is expected shortly.

From: Fraser Wilson
  <newsfraser@blancolioni.org>
Date: Wed, 05 Jun 2002
Subject: Safe Ada Unit Handling -- Macks
  0.1
Newsgroups: comp.lang.ada

Macks is a physical units preprocessor. Given a set of fundamental units, and other units derived from them, Macks produced Ada package specifications and bodies which implement these relationships in a type-safe way.

New features for this version: cartesian vectors, generated as records or arrays with arbitrary indexes/element names; vector operations: addition, subtraction, dot product, and cross product if the vector is three dimensional; unit subtypes; user header insertion; new operator: Unit / Unit -> Scalar; cosmetic fixes; source re-organisation and a Makefile; a manual!

You can grab it from
http://www.blancolioni.org/ada/macks/macks-0.1-src.tar.gz or
http://www.blancolioni.org/ada/macks/macks-0.1-src.zip

If you don't have texinfo installed, remove the 'doc' target from the top of the Makefile, and read the guide at
http://www.blancolioni.org/ada/macks/macks.html instead. [...], it can no longer be said that C++ has a resonable compile time implementation of typed units and Ada doesn't, which was really the point for me :-)

Cheddar - Real Time Scheduling Simulator

From: Pat Rogers
  <progers@classwide.com>
Date: Thu, 23 May 2002
Subject: new real-time tool written in Ada
Newsgroups: comp.lang.ada

There is an interesting schedulability analysis tool described in comp.realtime: "The LIMI team is pleased to announce the distribution of Cheddar, a free real time scheduling simulator mainly designed for educational purpose. Cheddar is now distributed under the GNU GPL license. Since this morning, Cheddar source code is available from: http://beru.univ-brest.fr/~singhoff/cheddar" [...]

[Cheddar provides automated services to check temporal constraints of real time tasks. -- dc]

When I downloaded the source, imagine my pleasant surprise seeing it is written in Ada!

UML Studio - Free Demo Version Generates Ada Code from UML

From: Pat Rogers
  <progers@classwide.com>
Date: Fri, 24 May 2002
Subject: Re: Free Ada UML tools
  Available?
Newsgroups: comp.lang.ada

Anybody know of any free tools that will allow me to model using UML and generate Ada code.

I did a lot of (uncompensated) work to make the UMLStudio tool from PragSoft generate what is IMHO appropriate Ada code from UML. It might be worth the time to see what it looks like, using their free demo version. I do not claim that my approaches are The One True Way of doing it, but at least I liked it! :-)
www.pragsoft.com

[Cf. also "PragSoft - UMLStudio Supports Ada" in AUJ 22-2 (June 2001), p.80. -- dc]

Dia & dia2code - Open Source Ada UML Tools

From: Preben Randhol
  <randhol+abuse@pvv.org>
Date: Thu, 23 May 2002
Organization: Norwegian university of
  science and technology
Subject: Re: Free Ada UML tools
  Available?
Newsgroups: comp.lang.ada

Use Dia and dia2code, the latter supports also Ada. Both are Free Software/Open Source.

Dia => http://www.lysator.liu.se/~alla/dia/

"Dia is a Gtk+ based diagram creation program released under the GPL license. Dia is designed to be much like the commercial Windows program 'Visio'. It can be used to draw many different kinds of diagrams. It currently has special objects to help draw entity relationship diagrams, UML diagrams, flowcharts, network diagrams, and simple circuits. It is also possible to add support for new shapes by writing simple XML files, using a subset of SVG to draw the shape.

It can load and save diagrams to a custom XML format (gzipped by default, to save space), can export diagrams to EPS or SVG formats and can print diagrams (including ones that span multiple pages)."

Dia2code =>
http://sourceforge.net/projects/dia2code

"This program generates code from an UML Dia Diagram. Currently supported languages are: Ada, C, C++, IDL, Java, PHP, Python, shapefiles and SQL."

From: byhoe@greenlime.com (Adrian Hoe)
Date: 23 May 2002
Subject: Re: Free Ada UML tools
  Available?
Newsgroups: comp.lang.ada

What a nice UML (and other) drawing tool! But still has plenty of room for improvement including the code generator.

[...] Still has to do a lot of hand work on the generated Ada code.

[The following day: -- dc]

I gave the Dia2code a quick hack last night. I was mainly trimming some of the code in generate_code_ada.c. It will generate nice spec and body files now. I will put it in my website so that it is available for download. At the same time, I will contact the Dia2code project manager for the update I made.

[And in response to "BTW, did you work a bit on the indentation of the code?": -- dc]

Yes, I did. I have also done some modification so that you can alter package extension to your liking rather than the fixed "_types" (eg. package client_types, client_types.ads). Just go to the generate_code_ada.c and change the [PACKAGE_EXT macro] to whatever you like.

Adrian Hoe, http://adrianhoe.com

ColdFrame - Ada Framework Code Generator for UML Tools

From: Simon Wright
  <simon@pushface.org>
Date: 24 May 2002
Subject: Re: Free Ada UML tools
  Available?
Newsgroups: comp.lang.ada

Anybody know of any free tools that will allow me to model using UML and generate Ada code.

I have a project (ColdFrame, http://www.pushface.org/coldframe/) which generates Ada framework code as a backend for a UML tool.

At the moment the only front-end supported is Rational Rose, which is_not_ free of course. The Gnome Office "Dia" sounds very interesting, though; the other

free UML tools are Java-based and (IME) a little unreliable. I had thought about PragSoft's offering, but have been very busy. All that's needed is to dump the model in a CF-specific XML format :-)

A word of warning: CF will generate a _lot_ of framework code for you because it assumes you don't want to clutter up your clean analysis model with framework cruft (collection classes etc). But for it to do this you have to invest a lot of effort into a model that is precise enough, and it needs you to put in more annotation than you might be used to (for example, you have to name each association and specify role names and multiplicities throughout).

From: Simon Wright
  <simon@pushface.org>
Date: 24 May 2002
Subject: Re: Free Ada UML tools
  Available?
Newsgroups: comp.lang.ada

What is it really that one expect of code from a UML model? I mean is it only the skeleton code or should it do more?

Well, I think it's up to the software architect! If your target is a little embedded box, you'll go one way, if it's something with an ODBC backend for persistence you'll do something else. But if the model describes the problem domain, it should be "true" for either. (This is a large claim, you often need to decorate/tag/colour the model to express particular design choices; of course a better design would tag the model descriptively and let the code generator figure out how best to translate it, but this is a hard problem).

The choice that the present CF architecture makes is appropriate (I think!) for a memory-resident VxWorks GNAT system. If you describe a class you get a whole raft of code to support the extent of the class (all the current instances) and even more if you have associations. It has state machine support as well (you just fill in the actions, which are operations of the class like any other).

[...] It is possible to generate a high proportion of the code needed, indeed if you write in an action specification language at a higher level than Ada you can get 100% translation of the same model into different target languages. See eg http://www.projtech.com/

Aonix have a framework-only parameterisable code generator (ACD). Some of the links from http://www.pushface.org/coldframe/resources.html are broken, but this one is good.

AtelierUML - UML Tool in Ada

From: Alexis Muller
  <Alexis.Muller@lifl.fr>
Date: Fri, 19 Jul 2002
Organization: LIFL
Subject: workshop UML in Ada
Newsgroups: comp.lang.ada

Hello, I translated a good part of meta-model UML (1.4) into Ada. I also made a graphic outline of interface. The goal being to obtain a free workshop UML of high quality allowing transformations of models (MDA) and to generate code (IDL, C++, Java, Ada...).

You can obtain the code on my site: [http://www.lifl.fr/~mullera/atelierUML/ -- dc]

ADI's Beacon Toolset Supports Ada Code Generation for Embedded Controllers

From: John Kern <jkern3@visteon.com>
Date: Wed, 29 May 2002
Organization: Visteon Coropration
Subject: Ada automatic code generation for
  Simulink models
Newsgroups: comp.lang.ada

[...] There seems to be a big push in the automotive industry for auto code generation of embedded controllers.

Too bad that we seem to be stuck with C.

From: rod@praxis-cs.co.uk (Rod Chapman)
Date: 30 May 2002
Subject: Re: Ada automatic code generation
  for Simulink models
Newsgroups: comp.lang.ada

I believe ADI's Beacon toolset has support for Ada (and SPARK... :-)) code generation from Simulink models. See www.adi.com

Rod Chapman, SPARK Team, Praxis Critical Systems

Ada and Cryptography

From: tuo_pe@yahoo.com (Tuomas P)
Date: 16 Jun 2002
Subject: Ada and cryptography
Newsgroups: comp.lang.ada

Is (can?) Ada being used in cryptographic applications?

If it is, could someone point me out to webpages that have cryptographic code in Ada? I have been reading Markus Kuhn's article on Ada95 ("Information for New Ada95 Programmers" [at http://www.cl.cam.ac.uk/~mgk25/ada.html -- dc]) and I have the impression that Ada might do well in that area (too!).

From: Florian Weimer
  <fw@deneb.enyo.de>
Date: Mon, 17 Jun 2002
Subject: Re: Ada and cryptography
Newsgroups: comp.lang.ada

Yes, it is. Read http://www.sparkada.com/downloads/ieeesw.pdf for an example.

[Article "Correctness by Construction: Developing a Commercial Secure System" by Anthony Hall and Roderick Chapman, Praxis Critical Systems, in IEEE Software, January/February 2002, pp.18-25. See also "Ada Success Story in IEEE Software Magazine" and "UK / Praxis CS – MULTOS Certification Authority for Smart Cards" in AUJ 23-1 (March 2002), p.32 and p.35. -- dc]

From: antonio_duran@hotmail.com
  (Antonio Duran)
Date: 17 Jun 2002
Subject: Re: Ada and cryptography
Newsgroups: comp.lang.ada

Yes it can. I'm trying to implement some cryptographic algorithms in Ada95 and you can see that work in: http://sourceforge.net/projects/adacf

[Cf. also "ACF - Ada Cryptographic Framework" in AUJ 23-1 (March 2002), p.18. -- dc]

[...] What you can find there is an implementation of some of the most important cryptographic hash (message digest) algorithms (MD2, MD4, MD5, SHA-1, RIPEMD-128, RIPEMD-160, TIGER, and HAVAL) and the test vectors for these algorithms. I'm currently working on a multiprecision natural number arithmetic package, cryptographic pseudorandom number generators (BBS) and some of the most important symmetric key ciphers (DES, 3-Way, Blowfish, and others). If you want the source code I could send you but be aware that is not production quality code (at least not yet).

From: Gisle Sælensminde
  <gisle@ii.uib.no>
Date: 17 Jun 2002 19:47:35 GMT
Organization: Institutt for Informatikk, UiB
Subject: Re: Ada and cryptography
Newsgroups: comp.lang.ada

Ada is in fact very well-suited for cryptography, especially symmetric crypto, as most of the data types and operators needed can be defined or are build in. For public-key crypto, big-number libraries are missing, but so are they doing in C, which is considered the "standard" language of cryptography. I have implemented the serpent algorithm in Ada myself: ftp://ftp.ii.uib.no/pub/ada/serpent.tar.gz [...]

[Cf. also "New Optimized Version of the Serpent Cipher" in AUJ 21-2 (July 2000), pp.100-101. -- dc]

From: David Marceau
  <davidmarceau@sympatico.ca>
Date: Mon, 17 Jun 2002
Subject: Re: Ada and cryptography
Newsgroups: comp.lang.ada

http://www.esat.kuleuven.ac.be/~rijmen/rijndael/rijndaelada.zip

[An Ada 95 implementation of the Advanced Encryption Standard (Rijndael). -- dc]

File Compression using Huffman Coding

From: Chryslid <chryslid@caramail.com>
Date: 17 Jun 2002
Subject: Re: code source pour construction arbre Huffman
Newsgroups: fr.comp.lang.ada

http://www.almink.com/school/huffman.html

[Also requires list_pkg.html and tree_pkg.html from same web-directory. -- dc]

Source Code Beautifiers for Ada

From: jim hopper <hopperj@macconnect.com>
Date: Fri, 19 Jul 2002
Subject: Re: source beautifier
Newsgroups: comp.lang.ada

Does a source code beautifier exist for Ada?

http://www.usafa.af.mil/dfcs/bios/mcc_html/adagide.html

AdaGIDE is a Windows IDE, but it also contains a standalone tool called reformat which does what you want. If you get the AdaGIDE source which is Windows specific, the reformat part of the source should compile on most platforms. [...] I am doing a GUI for it for Mac OS X (all in Ada using Apple's IDE and their GUI generator which we have tied together now).

[And from another message: -- dc]

There are a number of fixes to reformat available if you get the source from the AdaGIDE link I sent earlier. I was the one who extracted reformat from AdaGIDE and turned it into a tool I could use on about any platform. The older version has some subtle bugs that the new source code fixes thanks to Martin Carlisle and his folks! I have a few fixes as well that I haven't sent in yet so if it gives you problems send me an email and I will send them to you.

From: David Marceau <davidmarceau@sympatico.ca>
Date: Fri, 19 Jul 2002
Subject: Re: source beautifier
Newsgroups: comp.lang.ada

These are the alternatives that I know you might want to try [see elsewhere in this AUJ's news section for pointers and more information on most of these -- dc]:

1) Programmer's editor / emacs / vi / ghostscript. Have you tried opening your Ada file within emacs, select all and then indent-region (adjusts all the lines to be indented properly)? Have you tried printing from emacs? From what I

understand about source code beautifiers, emacs/ghostscript have both color/black and white and highlight keywords. It also does n-up printing meaning printing more than one source code page on one printed page. [...]

2) Use gnathtml. This produces a beautified html documentation of the source code along with an index. This also highlights keywords. Then you print what you want from your web browser.

3) Use adabrowse. This also produces a beautified html documentation of the source code along with an index. This one uses ASIS to produce it. This also highlights keywords. It does resemble the Java documentation style because that was the intent. Then you print what you want from your web browser.

4) Use RainCode. It not only beautifies but also code-metrics'fies among other things and combines the output of both in a well-organized html documentation with an index.

From: dewar@gnat.com (Robert Dewar)
Date: 20 Jul 2002
Subject: Re: source beautifier
Newsgroups: comp.lang.ada

There are several tools that perform this function. If you are a GNAT Pro user, then you should have a look at gnatpp, the GNAT Pro pretty printer, which is now available in a beta version to GNAT Pro users.

Robert Dewar, Ada Core Technologies

From: britt@acm.org (Britt Snodgrass)
Date: 20 Jul 2002 13:24:18 -0700
Subject: Re: source beautifier
Newsgroups: comp.lang.ada

Irvine Compiler Corporation provides their "ICC Ada95 Pretty Printer (ICCFMT)" for free at http://www.irvine.com/freebies.html It works pretty well although I remember it puts line breaks in a few strange places.

Resources for GNU-based Embedded Systems Development in Ada

From: William A. Gatliff <bgat@billgatliff.com>
Date: Friday, July 26, 2002
Subject: Embedded GNAT?
To: GNAT Discussion List <gnatlist@lyris.seas.gwu.edu>

I'm interested in writing an Ada application that runs on an embedded system with an RTOS, or perhaps no OS at all. I'm a big fan of GNU tools, and I've successfully used them in the past for embedded C/C++ development. I'm thinking that GNAT is the way to go here, but honestly, I'm such an Ada newbie that I don't even know where to begin. Any pointers to information would be much appreciated. [...]

From: Steve <steved94@attbi.com>
Date: Fri, 26 Jul 2002
Subject: Re: Embedded GNAT?
To: GNAT Discussion List <gnatlist@lyris.seas.gwu.edu>

Here is a link for a detail description of the GNU Ada Run-Time:
http://www.iuma.ulpgc.es/users/jmiranda/

[Cf. also "On-line GNU Ada Run-Time Book" further in this news section. -- dc]

You may find MarTE OS interesting:
http://marte.unican.es/

[Cf. also "MaRTE OS 1.0 - Minimal Real-Time Operating System for Embedded Applications" in AUJ 23-1 (March 2002), p.18. -- dc]

And then there's RTEMS:
http://www.rtems.com/

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

Since GCC now includes Ada as part of the main development tree, I've been expecting support for Ada in RTEMS on x86 to re-appear any time now. You can also get support for GNAT/VxWorks from ACT (www.gnat.com).

Ada-related Products

ACT - Wind River Partners with Smiths Aerospace and Ada Core Technologies to Deliver Tornado for Safety Critical Systems

URL:
http://www.gnat.com/texts/news/news_arinc.htm

Boeing Using New Safety-Critical Platform on C130AMP and 767 Tanker Transport Programs

Alameda, Calif., May 16, 2002 - Wind River Systems, Inc. (Nasdaq:WIND) and Ada Core Technologies today announced that they are teaming with Smiths Aerospace to deliver a commercial off-the-shelf (COTS) software solution compliant with the Aeronautical Radio, Inc. (ARINC) 653 avionics application software standard. The solution features Wind River's new Tornado(r) for Safety Critical Systems platform, supporting the development of a wide range of safety/mission critical embedded and desktop systems including aerospace avionics applications, military ground vehicle platforms, military and civil ground control stations as well as automotive and telecommunication devices. As part of the agreement, Smiths Aerospace is providing funding, market expertise and project management. Ada Core Technologies is providing support with its GNAT Pro environment, enabling developers working in the Ada

programming language to fully leverage this solution.

The Tornado for Safety Critical Systems platform combines Wind River's securely partitioned VxWorks(r) AE653 real-time operating system (RTOS), Tornado III integrated development environment and safety-critical features necessary to fully support robust partitioning of applications. Robust partitioning enables applications at different safety and criticality levels to be safely integrated onto a single processor - fully protected from each other - running on the same RTOS kernel. This saves significant time and certification costs without compromising safety when developing new or expanded safety critical systems.

"Smiths is applying the AE653 product to the Boeing C-130 Avionics Modernization Program, the Boeing 767 Tanker Transport, and other programs throughout the Smiths organization," said John Armendarez, Smiths Programs Director - Military Transport. "Collaboration with Wind River and Ada Core Technologies provides a mechanism to produce an FAA-certifiable commercial partitioned operating system that provides value to our customers by reducing the cost, size, power consumption and weight of our avionics products."

Smiths Aerospace is an industry leader in developing software applications and integrating systems utilizing software partitioned architectures. For the C-130 AMP and 767 Tanker Transport programs Smiths is integrating several avionics software applications, including the Smiths Flight Management System (FMS) and the Communication Management function, onto a single PowerPC processor hosting the Tornado for Safety Critical systems environment.

[...] Tornado for Safety Critical Systems is a complete COTS certification solution that provides developers with spatial and temporal partitioning in accordance with the ARINC653 standard. [...] The solution also enables avionics equipment manufacturers to mix and certify applications created in C and / or Ada languages, applications based on the POSIX and ARINC standards or existing Wind River VxWorks-based applications in an avionics system to different DO-178B levels -- fully protected from each other -- running on the same COTS-certifiable kernel.

[...] Ada Core Technologies is developing a set of software tools that supports various levels of safety-certified applications on Tornado for Safety Critical Systems. Its adaptation of the GNU Visual Debugger allows debugging of mixed language applications in Ada and C, as well as simultaneous debugging of several applications in different partitions. Since one of the main

economic benefits of a partitioned operating system is to support differing levels of certification in partitions, GNAT Pro will provide Ada runtime systems suitable for various levels, along with a full Ada runtime system. Each of these profiles provides a reasonable tradeoff between language restrictions and the cost of certification for a given level. The GNAT Pro compiler rejects the compilation of program units that violate a given profile. The compiler is also built to allow the specification of special purpose profiles so that only the corresponding runtime system need be built, without requiring compiler changes to enforce the new profile.

Tornado for Safety Critical Systems will be available for beta customers in August 2002. The platform will be made fully available in Feb. 2003 for Motorola's MPC74xx and MPC750 processors, with documentation to support certification to DO-178B, Level B. Documentation for support of DO-178B, Level A will be available in Dec. 2003.

Wind River is also providing a transition package for customers wishing to start early development. About Smiths Aerospace & About Wind River [see URL above -- dc].

About Ada Core Technologies, Inc.

Ada Core Technologies, Inc. (http://www.gnat.com/), a privately-held company founded in 1994 with major offices in New York City and Paris, produces and supports the GNAT family of open-source Ada 95 software development environments. Based on the GNU GCC technology, GNAT is available on more platforms than any other Ada compiler and is the only implementation of the complete Ada 95 language. GNAT Pro, the professional edition of the GNAT technology and the premier Ada development environment on the market, is used on enterprise-critical projects encompassing areas such as low-level communications control, high-integrity real-time applications, and large-scale distributed systems. [...]

Jim Barillo, Wind River,
jim.barillo@windriver.com
http://www.windriver.com
Nancy Cruz, Ada Core Technologies, Inc,
cruz@gnat.com http://www.gnat.com

ACT - SGI and Ada Core Technologies Team Up to Support Ada 95 Users

URL:
http://www.sgi.com/newsroom/press_releases/2002/june/ada.html

Ada Core Will Provide Direct Support to SGI IRIX OS-Based Customers for the Ada 95 Compiler

Mountain View, Calif. (June 26, 2002) – SGI (NYSE: SGI) and Ada Core Technologies (Ada Core), developer and maintainer of the Ada 95 GNAT Pro development environment today announced a new agreement for support. Effective July 1, 2002, the agreement will enable customers using systems based on the SGI IRIX operating system to obtain support for the Ada 95 development environment directly from Ada Core. This support agreement will allow SGI and Ada Core to leverage the strengths of both companies to serve their mutual customers more effectively.

"SGI is transitioning support to Ada Core in order to provide a closer relationship between SGI customers using the Ada 95 development environment and the Ada Core experts," said Wes Embry, western regional director, Ada Core Technologies, Inc.

Ada Core will provide its Premium level service to SGI customers who purchased Ada 95 compiler support contracts from Ada Core and to customers with existing SGI support contracts for the Ada 95 compiler. Customers will receive a rapid response to all problems submitted to the Ada Core technical support team.

"SGI has always been committed to providing our creative and technical customers with the best service in the industry," said Terry Oberdank, vice president of SGI Global Services. "This support agreement with Ada Core is in the best interest for our mutual customers. With Ada Core's exceptional development environment engineering expertise and support capabilities combined with SGI high-performance systems, the robust IRIX operating system and our own service experts, customers are getting the best of both worlds."

[...] Ada Core will continue to support SGI Ada 1.4.2 until the n32 product for SGI from Ada Core is released to SGI customers. Ada Core will support the o32 compiler which is part of SGI Ada 1.4.2 until July 1, 2003. The new n32 product from Ada Core will be based on GNATPro 3.15 and will contain new source level debugger tool called the GNAT Visual Debugger (GVD).

"This agreement is a continuation of the long relationship that SGI and Ada Core have enjoyed together. SGI was Ada Core's very first customer," said Embry. "Both parties are continuing to work together to ensure that a top quality Ada 95 development environment is available on IRIX."

For more information about SGI products, services or solutions visit
www.sgi.com
For more information about Ada 95 visit
www.gnat.com

About SGI. Celebrating its 20th year, SGI, also known as Silicon Graphics, Inc.,

is the world's leader in high-performance computing, visualization and the management of complex data. [...]

About Ada Core Technologies, Inc. [see previous news item -- dc] [...]

Aonix - ObjectAda for Windows Version 7.2.2 Update

URL:
  http://www.aonix.com/content/news/pr_5.24.02.html

ObjectAda for Windows Version 7.2.2 Update Now Available

Aonix, a leading provider of software development environments on Windows platforms, has released the next version of its object-oriented development environment, ObjectAda. This new release supports Windows 98, Windows NT, Windows 2000, and the embedded Intel/ETS Real-Time Win32 operating systems.

This is a point release that updates the last version (7.2.1) of ObjectAda for Windows and cross ETS. It consists of all patches and updates previously available for the product folded into a single full product release. This release also contains many other improvements not previously released. Information about specific updates can be found in the Release/Installation Notes contained on the product CD in the documents directory.

[...] ObjectAda for Windows version 7.2.2, Enterprise Edition, now comes with the AdaNav HTML source navigation and profiling tool set. This tool set is also included with the Project Pack version of ObjectAda Real-Time cross Intel/ETS. The addition of the AdaNav tool makes using ObjectAda far more productive. See the documents directory for user documentation describing AdaNav and Profiler tool set capabilities.

The new 7.2.2 Special Edition is now also available for download.
http://www.aonix.com/content/products/forms/oawin_dnld.html [...]

About Aonix
Aonix, a Gores Technology Group company, is a leading international software company with customers drawn from the Global 1000. The Critical Development Solutions (CDS) division produces Software through Pictures (StP), Architecture Component Development (ACD), TeleUSE, ObjectAda, AdaWorld and Raven. CDS products support the highest criticality levels of software design. [...] Aonix operates sales offices throughout North America and Europe in addition to a network of international distributors. For more information, visit www.aonix.com

Press Contacts: Greg Gicca, Director of Product Marketing, greg.gicca@aonix.com additional Product Information: info@aonix.com

Aonix - ObjectAda Intel/RAVEN 7.3 for Safety-Critical and Hard Real-Time Development

URL:
  http://www.aonix.com/content/news/pr_7.19.02.html

Aonix Announces the Newest Release of ObjectAda Intel/RAVEN 7.3 for Safety-Critical and Hard Real-Time Development

San Diego, California, July 19, 2002 - Aonix, a leading provider of embedded software development environments, is pleased to announce the release of Intel/RAVEN 7.3, a development environment for hard real-time and safety-critical development. Aonix is the leading supplier of certified (e.g., certifiable) run-time systems for the Ada 83 language and offers the only certified Ada 95 run-time system, certified to the highest certification levels, in the Raven product line. Raven meets the DO-178B Level-A required by the FAA for airborne systems. It also has mappings from this very strict safety-critical standard to others in high-speed rail, nuclear shutdown systems, etc. As other industries such as medical and automotive (to just name two) start to realize that their software systems can now effect human life, software safety is becoming a much more important issue.

Raven provides extremely fast and deterministic performance along with a small run-time-system footprint. The kernel is designed based on the industry standard Ravenscar profile.

This standard was originally developed to support both safety-critical and hard real-time system requirements. Thus, Raven is a valuable tool set for both safety-critical and general hard real-time embedded application development.

"This release of Intel/Raven provides a very large set of enhancements over the previous version. The Intel/Raven product inherits partitioned memory support from its PowerPC cousin, a new IDE/GUI, SCCI CM support, and a more advanced code generator from the ObjectAda for Windows product." states Greg Gicca, Product Manager for ObjectAda.

From the PowerPC product Intel/Raven 7.3 inherits support for the APEX industry standard for a partitioned memory mapped Operating Systems. With this support, Intel/Raven can be ported on top of such an OS to then support multiple safety levels on a single board. This greatly expands the

capabilities of Ada95 in safety-critical system development.

From the ObjectAda for Windows product, Raven inherits: a more powerful code generator for better program performance; many new debugging capabilities; a new Microsoft Visual Studio 6.0 IDE/GUI; and an SCCI (Source Code Control Interface) capability.

[Cf. also "Aonix - ObjectAda 7.2.1 for Windows" in AUJ 22-4 (December 2001), pp.207-208, for more on this SCCI capability, and the previous news item for general information on Aonix and contacts. -- dc]

Aonix - ObjectAda Patch Updates

From: owner-intel-objectada <owner-intel-objectada@sf.aonix.com>
Date: Thu, 01 Aug 2002
Subject: Intel-OA: New ObjectAda update 1102V722-U3
To: intel-objectada@sf.aonix.com

A new patch update to ObjectAda for Windows 7.2.2 (1102) is now available. The update download file and the Release Notes are available at http://www.aonix.com/content/support/ada/patches/objectada.html

Please see the Release Notes for more information.

Aonix Ada Support, adasupport@aonix.com

From: Aonix Ada Support <adasupport@sf.aonix.com>
Date: Tue, 20 Aug 2002
Subject: Unix-OA: Notification of update
To: unix-objectada@sf.aonix.com

There has been a patch update made recently to the following ObjectAda for UNIX product: ObjectAda for HP-UX 7.2 (7000), 7000V72-U2.

Please see the Release Notes available for the update for more information. The download files and Release Notes are available at http://www.aonix.com/content/support/ada/patches/objectada.html

Updates are available to customers under current maintenance. Contact your Aonix support office for the temporary download password.

ARTiSAN Software Tools - ARTiSAN brings UML to Ada Community

URL:
  http://www.ddci.com/news_vol3num3.shtml#3rd

Subject: DDC-I Online News, March 2002, Vol. 3 Issue 3

In the Summer of 2001, ARTiSAN Software Tools (www.artisansw.com) released the first version of their new Ada

code generation from UML. Real-time Studio has gained increasing acceptance among systems engineers and software designers in safety-related and mission critical software projects around the world. By bringing out an Ada code generator that supports full Ada generation, and also SPARK (the safe Ada subset), ARTiSAN indicated a new company focus on supporting safety related projects.

[Cf. also "ARTiSAN Software Tools - Ada Code Generator for UML Models in Real-time Studio" in AUJ 22-2 (June 2001), p.79. -- dc]

This summer, ARTiSAN will release full "round-trip" synchronization for Ada, and in particular offer a reliable and understandable translation from legacy Ada 83 code into UML.

As part of ARTiSAN's recognition that Ada is the language of choice for safety related projects, the synchronizer is being developed in cooperation with Praxis Critical Systems the birthplace of SPARK and supports the SPARK annotations for formal verification as well as the safe subset of Ada. (www.sparkada.com)

For more information on Real-time Studio or the Ada Synchronizer and its mapping to UML, please e-mail info@artisansw.com

ARTiSAN Software Tools - Real-time Studio Professional Selected by Saab Ericsson Space

URL:
http://www.artisansw.com/press/2002/Saab.asp

ARTiSAN's Real-time Studio Professional Selected by Saab Ericsson Space as Tool of Choice for Software Development

Cheltenham, UK -- July 2, 2002 -- ARTiSAN Software Tools, a global leader for UML-based, real-time systems and software modeling tools, today announced that Saab Ericsson Space, a world leader in the design, development, and manufacture of space electronics, has selected ARTiSAN's Real-time Studio Professional to support the development of mission-critical space and ground segment systems. The first projects using Real-time Studio involve developing sub-systems for the Galileo navigation system and the Herschel/Plank satellites. The Galileo navigation system will be the European equivalent of the American GPS, and the Herschel/Planck satellites are hoped to provide scientists with answers on how the universe was started and how stars and galaxies were born.

"After an extended evaluation of the top three UML real-time software development tools, it was obvious that ARTiSAN's Real-time Studio was the best fit for our needs," said Annalena

Johansson, Software Development Manager at Saab Ericsson Space. "Flexibility was key to our choice of RtS; many of our projects share common designs and are implemented in a mix of languages (Ada and C), so we needed a tool that would allow teams to share models in a controlled manner and that would support multiple languages in a single model. [...]"

Jeremy Goulding, President and CEO of ARTiSAN commented: "After in-depth evaluations, Real-time Studio has once again shown itself to be the best competitive offering on the market. Our strategic support of Ada within our UML environment, offering now reverse and forward engineering for Ada 83 and Ada 95, and our continuing commitment to provide good tool support for distributed software development teams has paid dividends with this six-figure purchase."

Real-time Studio was also chosen in parallel by Austrian Aerospace, a wholly-owned subsidiary of Saab Ericsson Space, after this extended evaluation.

About ARTiSAN Software Tools [see URL above -- dc].
ARTiSAN Software Tools, Inc., founded in March 1997, is privately held with headquarters in Portland, Oregon and Cheltenham, United Kingdom. The company has regional sales offices and distributors throughout the world. For more information, visit:
www.artisansw.com

About Saab Ericsson Space
Saab Ericsson Space is an international supplier of qualified space equipment. The company develops and manufactures computers, microwave electronics and antenna systems, guidance and separation systems and thermal hardware for launch vehicles and satellites, and is the world leader in several of its product areas. The headquarters is located in Göteborg, Sweden and there are subsidiaries in Austria and in the USA. The company employs 660 people. For more information, please visit:
www.space.se [...]

Press Contact: ARTiSAN Software Tools, Richard Gastwirt,
richard.gastwirt@artisansw.com

DDC-I - SCORE Suite for OSE RTOS for Safety Critical Software Development

URL:
http://www.ddci.com/news_vol3num6.shtml
Subject: DDC-I Online News, June 2002, Vol. 3 Issue 6

[...] SCORE for OSE: A Strong Platform for Safety Critical Software Development, By Peder Møller, DDC-I A/S

The integration of the SCORE (Safety-Critical, Object-oriented, Real-time Embedded) suite of programming and testing tools with the OSE RTOS (Real-Time Operating System) from OSE Systems combines a compiler and an RTOS that are both designed for use in high-integrity embedded systems. For the first time, SCOREs multi-language support provides the application developer the ability to utilize both Ada and C for developing OSE applications.

The combination of Ada's reliability provided through strong typing and rigorous checks and the built-in safety features of the certified OSE RTOS, offers a strong platform for safety-critical software development. The SCORE IDE with OSE support runs on Solaris and Windows NT platforms and can be used for generating PowerPC cross applications or for generating Solaris native applications running on the OSE SoftKernel. SCOREs OSE support encompasses both compiler, GUI, and Multi-Language Debugger (MLD) support.

The OSE RTOS is designed for use in fault-tolerant systems. It has been certified according to the international IEC 61508 safety standard widely used in industrial systems and is certifiable according to the DO-178B standard for avionics systems. The OSE kernel is fully preemptive with deterministic real-time behavior.

OSE achieves a high level of safety by restricting interprocess communication to a safe and flexible message passing scheme that avoids the need for shared memory. The efficient encapsulation of OSE processes makes it possible to utilize the OSE MMU support to get full memory protection between processes.

Furthermore, the OSE kernel contains extensive internal error checks and a flexible error handling mechanism. The safety guidelines in the OSE documentation help the application developer to employ OSE for producing highly reliable applications. OSE includes numerous embedded communication products that make it particularly well suited for distributed systems. An OSE application can for instance be configured with OSE processes running on multiple CPUs and multiple boards communicating through TCP/IP, and the OSE signal communication can be configured so that it is transparent to the OSE processes whether other processes are local or remote.

OSE has preconfigured Board Support Packages (BSPs) for a wide range of PowerPC target boards. The OSE BSPs contain all the required board specific code for handling the board initialization and drivers for interrupt controllers, serial communication units, and Ethernet devices.

SCORE supports compilation, linking, and debugging of applications running on top of the OSE RTOS. The SCORE OSE applications can be written in Ada or C, or in a mixture of the two languages. Both Ada and C code may use OSE processes to support multi-process applications.

The SCORE system has a clean and board-independent interface to OSE which is provided through SCOREs OSE UCC. This interface is the link between the SCORE target libraries, primarily the SCORE Run-Time System, and OSE. The interface has been designed to ensure that hardware and board specific details are entirely within the OSE domain. This means that the SCORE target code is 100 % board independent and that SCORE OSE applications can run on any target that has an OSE BSP.

The Ada root library has been extended with interface packages containing Ada bindings to the OSE system calls providing a seamless interface from Ada to OSE while maintaining the safety of Ada's inherent type checking. A special variant of the OSE interface package containing only the IEC 61508 and DO-178B certified subset of the OSE system calls allows the user to check at compile-time whether the application is restricted to use this subset.

[...] The SCORE OSE integration supplements the OSE installations application examples with new examples of pure Ada and mixed Ada and C applications which allow the user a quick way to learn how these applications can be configured and how the Ada interface to OSE can be utilized. [...]

DDC-I - DACS Multi-Application Programming with Paging

URL:
 http://www.ddci.com/news_vol3num7.shtml
Subject: DDC-I Online News, August 2002, Vol. 3 Issue 7

[...] By Thørkil B. Rassmussen, Senior Software Engineer, DDC-I A/S

DDC-I, in cooperation with a large U.S. customer, has created an extension to the traditional DACS-80x86 cross compiler environment for 32-bit, flat mode Pentium targets that introduces the ability to have more than one application executing on the bare target at any time.

The traditional cross environment consists of a single, monolithic application (program), linked with the proper Run-Time System (RTS) and User Configurable Code (UCC), loaded onto the target in one of several ways, and then executed, possibly indefinitely. Such an application can be debugged, when it is loaded by the debugger.

It is desirable to split this one application into several smaller, simpler applications, perhaps with different priorities, capabilities or rights. This can be modelled to some extent assuming a set of tasks for each application, and providing each with proper attributes. Often, however the smaller applications have some parts in common with each other. This led to the idea of allowing the common parts to be shared by more than one application.

[...] The multi-application system allows for a late configuration of the set of applications for a target board for a given task or mission, and it may even be decided to leave out some less used applications initially. The user may decide to load inspection applications later that can check proper system function, or may break into the target to look at it, with a later withdrawal. This gives a great deal of versatility, as trusted applications can remain as they are in many configurations and need not be linked again and again as part of other applications. The multi-target capability also makes debugging of complex target board setups possible.

DDC-I's multi-application programming with paging (MAPP) extension can be of great advantage to customers with complex needs, involving several target configurations and/or several boards, and requiring full symbolic debugger support all the way. [...]

IBM - FLAVERS: Finite State Verification Tool for Ada and Java

From: Jim Rogers
 <jimmaureenrogers@worldnet.att.net>
Date: Thu, 28 Feb 2002
Subject: IBM Still Remembers Ada
Newsgroups: comp.lang.ada

I just ran across a reference to the following article:
http://www.research.ibm.com/journal/sj/411/cobleigh.html

Note that IBM has developed this finite state verification tool to work with Ada and Java.

URL:
 http://www.research.ibm.com/journal/sj/411/cobleigh.html
IBM Systems Journal, Volume 41, Number 1, 2002, Software Testing and Verification

FLAVERS: A finite state verification technique for software systems, by J. M. Cobleigh, L. A. Clarke, and L. J. Osterweil

[Some extracts. -- dc]

Software systems are increasing in size and complexity and, subsequently, are becoming ever more difficult to validate. Finite state verification (FSV) has been gaining credibility and attention as an alternative to testing and to formal

verification approaches based on theorem proving. There has recently been a great deal of excitement about the potential for FSV approaches to prove properties about hardware descriptions but, for the most part, these approaches do not scale adequately to handle the complexity usually found in software. In this paper, we describe an FSV approach that creates a compact and conservative, but imprecise, model of the system being analyzed, and then assists the analyst in adding additional details as guided by previous analysis results. This paper describes this approach and a prototype implementation called FLAVERS, presents a detailed example, and then provides some experimental results demonstrating scalability.

[...] FLAVERS, FLow Analysis for VERification of Systems, has been developed for application to the analysis of systems written in the Ada or Java languages. [...] Our FLAVERS prototype for analyzing programs written in Ada is considerably more mature than the prototype for Java code. Therefore all the results presented in this section are based on FLAVERS/Ada. [...]

IPL - IPL to Support Green Hills Software's MULTI Development Tools

From: Sue Plews <sue@pinnacle-marketing.com>
Date: Tue, 23 Jul 2002
Subject: IPL to Support Green Hills Software's Leading MULTI Development Tools
To: <Dirk.Craeynest@offis.be>

Press Release, 19th July 2002

IPL, the high integrity software testing tool company has joined Green Hills Software's Third Party Program. As a Program member, IPL will integrate its Cantata++ and AdaTEST tools with Green Hills Software's MULTI Integrated Development Environment (IDE). The integrated solution will make it even easier for software developers to produce high-quality certifiably tested code.

"The Cantata++ and AdaTEST products are designed for the verification of high-reliability software," said Ian Gilchrist, IPL's Software Products Consultant. "Giving our customers world-class software support is important to us and MULTI is a major IDE partner."

"Customers want professional end-to-end tools when creating their code for high-integrity software products," said Christopher Smith, Director of Marketing at Green Hills European Operations. "IPL's products and MULTI combine to give customers a mature, feature-rich development environment which will enhance their ability to produce reliable software components. We're pleased with

IPL's support plans and welcome them as a Program partner."

IPL's tools are the premier products for the unit and integration testing phases of the development lifecycle. Cantata++ and AdaTEST produce thorough host and target tests of C, C++ and Ada, with evidence of code coverage and conformance to coding and complexity standards.

MULTI 2000, Optimizing C/C++ compiler, ThreadX and INTEGRITY real-time operating systems, along with the Green Hills Probe make it easy to produce properly tested code components for high-performance microprocessors, including PowerPC, ARM, MIPS and Intel. Featuring a window-oriented text editor, RTOS-aware source-level debugger, and graphical program builder, MULTI 2000 also includes a run-time error checker, version control system, and performance profiler, as well as an instruction set simulator that allows programmers to develop their code on a PC or workstation without the need for the target hardware.

More on IPL

IPL was founded in 1979, develops leading edge testing tools that are used by software developers around the world. The current range of software testing products was first launched in 1992, and has since been expanded by the addition of new products, new facilities and newly supported platforms. The IPL tools are widely known and respected amongst the world's developers of high-integrity software. These include those working to all of the recognized software-safety standards, and many of those for whom software reliability is simply a 'must' for commercial reasons.

More on Green Hills Software, Inc.

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

Green Hills Software is headquartered in Santa Barbara, CA, with European headquarters in the United Kingdom. For more information on Green Hills Software products, please email inquiries to:
sales@ghs.com
or visit
http://www.ghs.com [...]

For more information contact: IPL, Ian Gilchrist, ian.gilchrist@iplbath.com [...]

PegaSoft - TextTools 1.0.5 Console Development Package

From: PegaSoft Canada - ALT Drop Box
   <adalin-l@tiamet.vaxxine.com>
Date: Wed, 22 May 2002
Subject: TextTools 1.0.5 Released
To: GNAT Discussion List
   <gnatlist@lyris.seas.gwu.edu>

My TextTools library has been updated to work with NCURSES 5 (as supplied with Red Hat 7.1) and GNAT 3.14.

[Cf. also AUJ 22-3 (September 2001), p.146. -- dc]

The new version is available for download from: http://www.pegasoft.ca/tt.html

Ken O. Burtch, http://www.pegasoft.ca ken@tiamet.vaxxine.com, Pegasoft, Jordan Station, ON, Canada

Praxis Critical Systems - SPARK Toolset 6.1

From: rod@praxis-cs.co.uk (Rod Chapman)
Date: 11 Jul 2002
Subject: ANNOUNCE: SPARK toolset 6.1
   now available
Newsgroups:
   comp.lang.ada,comp.realtime,comp.progr
   amming,comp.software-eng

Praxis Critical Systems is pleased to announce the immediate availability of release 6.1 of the SPARK language and the SPARK toolset.

[...] A new release for readers of John Barnes' "SPARK Book" will be available soon.

[Cf. also AUJ 22-4 (December 2001), p.210. -- dc] The full release note is available on-line at www.sparkada.com

The full definition of SPARK ("The SPARK Report") is also available to interested parties upon request.

Release 6.1 of the SPARK language adds:

* A useful subset of Ada95's tagged types are now permitted in SPARK95. Inheritance and extension of tagged types are allowed, as is the overriding of basic operations. Dispatching operations are not permitted.

* A new "type assertion" annotation can be used to specify the predefined base type from which an integer type declaration is derived. A complementary configuration-file mechanism allows a user to specify the details of packages Standard and System for a particular implementation. When used together, these facilities greatly improve the Simplifier's ability to discharge run-time exceptions verification conditions arising from Overflow_Check in both SPARK83 and SPARK95.

* Both full-range and constrained subtypes of modular types are now permitted in SPARK95.

There are also significant enhancements to the Examiner, Simplifier and Checker with this release. Highlights include:

* When the configuration file mechanism is used to specify packages Standard and System, the Examiner can check a number of additional static semantic rules of Ada. This also improves VC generation and simplification where the ranges of the predefined base types are required.

* Simplifier version 2.04 ships on Windows and Solaris platforms with this release. This includes improved proof tactics for enumerated inequalities, common SPARK idioms such as a "for" loop over an enumerated range, and simplification of quantified expressions. The performance of the simplifier has also been greatly improved on large VCs.

* Checker 1.47 ships on Windows and Solaris with this release. This includes extended rule families for dealing with modular arithmetic, bitwise operators and enumerated types.

This release re-inforces SPARK's position as the leading language subset and static analysis technology for the construction of high-integrity, critical software.

Please email us for more information at sparkinfo@praxis-cs.co.uk
or see
www.sparkada.com

Note: The SPARK programming language is not sponsored by or affiliated with SPARC International Inc and is not based on SPARC(tm) architecture.

RainCode - XMLBooster for Ada Released

URL: http://www.xmlbooster.com/

XMLBooster: the ultimate solution in XML application integration

* Application integration of XML data cannot get any simpler: instead of dealing with sophisticated APIs (such as DOM or SAX), use a convenient data structure, generated to suit your specific purpose, in the language of your choice.

* Supports numerous languages, including C, C++, COBOL, Delphi, Java and Ada.

* Statically generates application-specific parsers, to be included royalty-free in your application.

* Generates convenient structural documentation to describe the structure of the XML files or messages your application supports.

* Supports translators to and from DTD's, and from Schemas.

* Generates a GUI to edit your specific XML files. This GUI can be integrated in your application. [...]

URL: http://www.xmlbooster.com/Ada.htm

XMLBooster's Ada Code generator

* Generates a parser (to convert an XML input to a valid Ada data structure) as well as an unparser (to convert the Ada data structure back to an XML stream).

* Generates standard Ada95 code, that can be compiled on virtually any platform, including NT, Linux, Unix, etc.

* Inheritance in terms of elements is mapped onto equivalent inheritance relationships for the corresponding generated objects.

* Performs error recovery when dealing with a recoverable error, and raises an Ada exception in case of a non-recoverable error.

* Fully supports regular expressions: a data element can be attached to a regular expression, which will be used to validate the data fetched from the incoming XML message at parse time.

* Aggressively tested memory management strategy, in order to guarantee stability of memory usage even when millions of XML messages have been processed. [...]

URL: http://www.xmlbooster.com/profile.htm

The company behind XMLBooster: XMLBooster is developed, maintained and marketed by RainCode, a software company based in Brussels (Belgium) and which is specialized in object-oriented technology, compiler design and mission-critical real-time systems.

Contact us: info@xmlbooster.com , RainCode, Brussels, Belgium

RainCode - Evaluation Version of RainCode for Ada Available

From: Deborah Torrekens
*   <deborah@phidani.be>*
Date: Thu, 18 Jul 2002
Subject: RainCode for Ada
Newsgroups: comp.lang.ada

http://www.raincode.com
RainCode for Ada - Press Release - July 2002

RainCode for Ada is a quality control technology that operates on large amounts of existing Ada code, both legacy or during development. RainCode detects, counts, and measures non-trivial elements in your Ada code, and it can take any corrective or preventive action in it.

[Cf. also "RainCode Corp. - RainCode for Ada" in AUJ 22-4 (December 2001), p.210. -- dc]

Do you want to see RainCode at work? It is simple and easy: just download the demonstration / evaluation version of the RainCode Engine for Ada.
http://www.raincode.com/Downloads.htm

The RainCode dynamic step-by-step demo is a Wizard-like environment, which allows you to see how RainCode actually works, without having to go through heavy tutorials or specific workshops.

When downloading this demo you actually get a version of RainCode running on your computer, which shows you much more than a simple slide-show presentation. This demo presents you some of the most essential RainCode functions, applied to Ada-source samples taken from GNAT. Make the demo run, and RainCode executes itself before your very eyes. This program includes: the source code on which RainCode is applied, the result of RainCode's work, for each step, an explanation of what it does, the RainCode scripts.

In less than 5 minutes, you can see how RainCode works, and some of the things it does. If you want to investigate further what can be done with the RainCode Engine, the evaluation part will allow you to write all the scripts you want and apply them on a set of given sources.

Rational - Apex 4.2.0b, TestMate 4.2.0b, Ada Analyzer 4.2.1 for Sun Solaris

From: Greg Bek <gab@Rational.Com>
Subject: Rational Apex 4.2.0b, Rational TestMate 4.2.0b, Ada Analyzer 4.2.1 for Sun Solaris are available by FTP
To: Apex Announcements <apex-announcements@Rational.Com>

[Extracted from postings on Thu, 22 Aug 2002. In all Rational URLs below, substitute:
<apex> by <ftp>/apex;
<testmate> by <ftp>/testmate;
<analyzer> by <ftp>/ada_analyzer;
<rel> by releases/sol;
<doc> by documents/unix;
<ftp> by ftp://ftp.rational.com/public
-- dc]

This release is pending Generally Available (GA) status as it goes through the final steps of the manufacturing process. We anticipate that this will be complete within the next 30 days. Once this release reaches GA status, it will be available for shipping. Until then, it is being provided on this FTP server for immediate access. Follow this link for Rational Apex or Rational TestMate or Ada Analyzer download and installation instructions. [...]
<ftp>/.standard.msgs/install_instructions.html

Product: Rational Apex. Version: 4.2.0b.
Platform: Sun Solaris.
URL: <apex>/<rel>/apex.4.2.0b
Release Note
<apex>/<doc>/release_note.4.2.0b.dir/release_noteTOC.html
<apex>/<doc>/release_note.4.2.0b.ps.Z
Install Guide:
<apex>/<doc>/install_guide.4.2.0b.dir/igTOC.html
<apex>/<doc>/install_guide.4.2.0b.ps.Z

Product: Rational TestMate. Version: 4.2.0b. Platform: Sun Solaris.
URL: <testmate>/<rel>/testmate.4.2.0b
Release Note:
<testmate>/<doc>/testmate_release_note.4.2.0b.dir/testmate_noteTOC.html
<testmate>/<doc>/testmate_release_note.4.2.0b.ps.Z

Product: Ada Analyzer. Version: 4.2.1. Platform: Sun Solaris.
URL: <analyzer>/<rel>/ada_analyzer.4.2.1
Release Note:
<analyzer>/<doc>/release_note.4.2.1.dir/AA_Release_Note.4.2.1.html
<analyzer>/<doc>/release_note.4.2.1.ps.Z

Rational - Apex Embedded 4.2.0 for Sun Solaris to PowerPC Family

From: Glenn, Eddie <cav@Rational.Com>
Subject: Rational Apex Embedded, Sun Solaris to PowerPC Family for Tornado, LynxOS, and Rational Exec, version 4.2.0 is available by FTP
To: <apex-announcements@Rational.Com>

[Extracted from postings on Thu, 22 Aug 2002 and Tue, 27 Aug 2002. In all Rational URLs below, substitute:
<rel> by <ftp>/apex_cross/releases/sol/power;
<doc> by <ftp>/apex_cross/documents/unix;
<ftp> by ftp://ftp.rational.com/public
-- dc]

Product: Rational Apex Embedded for Tornado (vw), LynxOS (lx) & Rational Exec (rt). Version: 4.2.0.
Platform: Sun Solaris to PowerPC.
URL: <rel>/vw/4.2.0, <rel>/lx/4.2.0 & <rel>/rt/4.2.0
This release is pending Generally Available (GA) status [see above -- dc].

Follow this link for Rational Apex for Tornado, LynxOS & Rational Exec download and installation instructions. [...]
<ftp>/.standard.msgs/install_instructions.html

Release Note for VxWorks:
<doc>/VxWorks/relnote.4.2.dir/vxworks_relnote_ppc/vxworks_release_noteTOC.html
<doc>/VxWorks/relnote.vxworks.ppc.4.2.ps.Z
Release Note for LynxOS:
<doc>/LynxOS/relnote.4.2.dir/lynx_relnote/lynx_release_noteTOC.html<doc>/LynxOS/relnote.lynx.4.2.ps.Z
Release Note for Rational Exec:
<doc>/Rexec/relnote.4.2.dir/rexec_relnote_ppc/rexec_release_noteTOC.html
<doc>/Rexec/relnote.rexec.ppc.4.2.ps.Z

R.R.Software - Janus/Ada 83 for MS-DOS Still Available

From: Randy Brukardt
    <randy@rrsoftware.com>
Date: Wed, 20 Mar 2002
Subject: Re: periodicity
Newsgroups: comp.lang.ada

Janus/Ada 83 for MS-DOS is still available (at least until the manuals run out or we move again...); all it requires is a 640K MS-DOS machine - even an 8086 will work. That was a fully validated Ada 83 compiler (ACVC 1.11, the last Ada 83 one).

The price [for the Janus/Ada 83 Compiler for 80X86 MS-DOS] is still $129 (see www.rrsoftware.com). We'd charge less if someone wanted a bunch (we don't have any manufactured ones left, so there is a cost to making one on demand).

From: Randy Brukardt
    <randy@rrsoftware.com>
Date: Wed, 12 Jun 2002
Subject: Re: Embedded Ada Development
    Tools
Newsgroups: comp.lang.ada

Is there any $500 or less Ada compiler for any small embedded system?

Janus/Ada 83 [Professional Development System -- dc] for MS-DOS is still $500. It has been used for embedded 186-based systems using an embedded MS-DOS. (The full bare-machine version costs a lot more than $500, though). Don't know if this is small enough for you, though.

Top Graph'X - CORBA Middleware Components in Ada

URL:
    http://www.ddci.com/news_vol3num5.shtml#3rd
Subject: DDC-I Online News, May 2002,
    Vol. 3 Issue 5

[...] 3rd Party Update: Top Graph'X Targets Embedded Systems

Top Graph'X has been developing high quality middleware components in Ada for more than a decade. Their initial product, XInAda, is the only X-Windows environment that correctly supports Ada tasking. Recently Top Graph'X has been focusing on various Object Management Group (OMG) specifications. These products facilitate object-oriented distributed programming, using the Common Object Request Broker Architecture (CORBA). In 1997 Top Graph'X was one of the first ORB vendors to support CORBA application development in Ada.

During the 2001 fiscal year Top Graph'X initiated a vigorous development effort, culminating first quarter, 2002, to bring new products to the field. They now

support CORBA applications developed in Ada95, Java, and C++ as well as real-time CORBA. This makes high quality, reliable CORBA technology available for almost any environment. Fourth quarter 2002 will bring OrbRiver-Critical, providing effective, trustworthy CORBA technology for embedded systems applications.

OrbRiver-Critical implements the minimum CORBA, CORBA messaging, and real-time CORBA OMG specifications. It is based on standard OrbRiver technology, and provides a complete set of configuration parameters to tailor resources to your precise needs. These features, together with pluggable transport, inherent reliability, and high performance, provide an unprecedented opportunity for using middleware to reduce the cost and complexity of embedded systems development.

http://www.topgraphx.com
Europe & US
info@topgraphx.com [...]

Vector Software - VectorCast Automated Test Tools

URL:
    http://www.ddci.com/news_vol3num7.shtml#3rd
Subject: DDC-I Online News, August 2002,
    Vol. 3 Issue 7

[...] 3rd Party Update: Vector Software, The "Next Generation" of Intelligent Test Tools.

Vector Software, Inc. is a leading independent provider of automated test tools for software developers. Established in 1989 as a consulting and service organization, Vector's product focus is to empower software professionals to deliver the highest quality software in the least amount of time. Vector's "VectorCAST" line of products reduce the burden placed on individual developers by automating and standardizing application component level testing. This innovative technology developed by Vector represents the "next generation" of intelligent test tools. The tools support Ada, C/ C++ and Embedded C++ (EC++).

VectorCast has been integrated with DDC-I's SCORE (Safety Critical Object-oriented Real time Embedded) and DACS (DDC-I Ada Compiler System) toolsets. [...]

Vector Software, Inc., North Kingstown, RI, info@vectors.com

Ada and Linux

Debian GNU/Linux and GNAT

From: Samuel Tardieu <sam@ada-
    france.org>
Date: Mon, 03 Jun 2002
Subject: [ada-france] Debian GNU/Linux et
    GNAT
To: ada-france@ada-france.org

[Translated from French: -- dc]

Information for the users of Debian GNU/Linux: GNAT 3.14p is still included in the "gnat" package (for INTEL, Sparc and PowerPC), but, new, GNAT coming with GCC 3.1 is available by installing the package called "gnat-3.1" (thanks to the persons in charge for GCC for Debian). This version is not as stable as GNAT 3.14p, but will allow you to use a more recent GNAT if you wish.

Ada and Microsoft

AIDE - Ada Instant Development Environment for Windows

From: Stéphane Rivière
    <stephane@rochebrune.org>
Date: Thu, 09 May 2002
Subject: [AIDE] Lettre d'information n°1 du
    09/05/02
Newsgroups: fr.comp.lang.ada

[Cf. also same topic in AUJ 23-1 (March 2002), p.29. Extracts translated from French: -- dc]

The first public version was finally released [...] and is available on the net (in its entirety): http://eig.unige.ch/lii/Aide.htm and by mail: S. Rivière, 3, Imp. of Chasseries, F-17550 Dolus d'Oléron, France.

[...] "AIDE" was tested successfully under Windows 9x, Windows NT 4 and Windows 2000. [...]

[From the web-page above: -- dc]

"AIDE is an Ada/GtkAda/GVD environment under Windows, up to date and directly usable by a simple copy. In this manner, you will have the possibility of familiarizing you with this extraordinary language without spending hours to configure your development environment."

Prebuilt GCC 3.1 with Ada Package for Windows

From: Sune Falck <nospam@nospam.com>
Date: Mon, 20 May 2002
Subject: Prebuilt version of gcc 3.1 with
    Ada as an additional package for windows
Newsgroups: comp.lang.ada

The mingw project www.mingw.org has made available prebuilt gcc 3.1 core, gcc

AdaGIDE Mirror on Ada-Belgium Ftp-archive

From: Dirk Craeynest
*   <Dirk.Craeynest@cs.kuleuven.ac.be>*
Date: Wed, 22 May 2002
Subject: Re: AdaGIDE mirror
To: ada-france@ada-france.org

Could you imagine to establish an additional mirror for
ftp://ftp.usafa.af.mil/pub/dfcs/carlisle/adagide/
http://www.usafa.af.mil/dfcs/bios/mcc_html/adagide.html
under the condition that Martin Carlisle allows this?

I have added an extra mirror for AdaGide to the mirrors on our Ada-Belgium ftp-archive. Check out:
ftp://ftp.cs.kuleuven.ac.be/pub/Ada-Belgium/mirrors/adagide/

Claw Graphical Demo - Chaotic Lorentz Attractor

From: François Fabien
*   <fr.fabien@infonie.fr>*
Date: Sat, 22 Jun 2002
Subject: Re: Ada + Web + CGI
Newsgroups: comp.lang.ada

If you [...] choose to use the Windows interface a sound choice is to use RR Software's Claw. I have done a small demo with it on the chaotic Lorentz attractor with drawing at
http://membres.lycos.fr/frfabien/

Scintilla Programmers Editor and GWindows

From: David Botton <David@Botton.com>
Date: Mon, 24 Jun 2002
Subject: Scintilla Programmers Editor and GWindows
Newsgroups: comp.lang.ada

Since GWindows now includes complete bindings to Scintilla, an excellent programmers editor with support for Ada, I sent the text below off to the Scintilla user group. I though it would be of interest here as well. I ultimately plan on making sure every Delphi/C++ Builder and VB/VC++ programmer knows that GNavi (The Open Source Delphi/VB) exists ☺

As a test of Scintilla, I embedded with little work the Scintilla control in to my GNavi "proof of concept" and already have a somewhat functional Ada IDE :-)

I hope to add a little more functionality (spec. some basic code generation for GUI layouts) to the GNavi "proof of concept" (meaning that I just tried out some ideas in the code and this is very much not what GNavi will ultimately be) and I'll make it available for people to play with :-)

3.1 Ada, binutils and other support files at

--

As part of a larger product called GWindows - A rapid development Open Source GUI framework for Windows with advanced features like ActiveX and Database support and more, which is yet part of a larger product called GNavi - An open source Delphi / VB like environment for Windows (but using a language reliable enough to control the Boeing 777 with your family on board ☺

I have created a compete binding to Scintilla as part of the Gwindows framework and plan on using it (and already have for some very raw proofs of concept) as the editor for GNavi.

http://www.adapower.com/gwindows

Ada is an object-oriented general purpose programming language with Pascal roots and known as the international language of software engineering. It has built-in concurrency, built-in exception handling, built-in generic templates, built-in distributed execution, standard interfaces to other programming languages and libraries, compiles to Java Byte Code and just about any OS under the sun, and a rich set of primitives for getting at and putting to work all that expertise you've ever desired to embed in your software.

The GNavi project uses GNAT, the gcc Ada compiler. Making it a complete open source replacement for advanced Win32 development on all levels.

ActiveX in Ada

From: Andreas <Amun_Ra72@gmx.net>
Date: Mon, 1 Jul 2002
Subject: ActiveX in Ada?
Newsgroups: comp.lang.ada

Since it's possible to do DLL generation with Ada, the question now is: is it possible to create ActiveX (OCX) controls in Ada? The question specifically goes for Aonix Object Ada, since it already can incorporate/use existing ActiveX controls. I would really like to leave the VB/VC++ realm for creating OCXs.

From: David Botton <David@Botton.com>
Date: Tue, 2 Jul 2002
Subject: Re: ActiveX in Ada?
Newsgroups: comp.lang.ada

David Marceau
<davidmarceau@sympatico.ca> wrote
http://www.adapower.com/gnatcom/

chris.danx <danx@ntlworld.com> wrote

Isn't that a GNAT only solution?

The questioner thought (wrongly) that perhaps Aonix Object Ada would be the only solution since it supports ActiveX. [...] Actually, GNATCOM - COM/DCOM/ActiveX/COM+/MTS/OLE 2/OLE Automation support for Ada [with

http://sourceforge.net/project/showfiles.php?group_id=2435 [...]

GNAT] - allows full use and creation of all the above buz words and much more :-) If you use GWindows with GNATCOM you get the best support for Graphical ActiveX use and development available....

In fact I created a GUI ActiveX control over year and half ago just using GNATCOM. I decided to write GWindows in a large part to fill out areas of ActiveX control programming (there still are many great things to come in that area as the GNavi - the Open Source Delphi/VB alternative - project is starting to move forward).

Currently GNATCOM and GWindows are only on GNAT, but there is no technical reason at this point to my knowledge that it could not be ported to any other compiler.

[Greg Bek <gbek@bigpond.net.au> responded: -- dc] Rational Apex on NT supports the GNATCOM bindings.

From: David Botton <David@Botton.com>
Date: Wed, 3 Jul 2002
Subject: Re: ActiveX in Ada?
Newsgroups: comp.lang.ada

[...] The DLLs can actually be regular DLLs provided the author included a typelib, but in practice, I've never seen that. A bit of a surprise as it allows VB to also instantly bind to the DLL. Another common extension for Activex is OCX instead of DLL.

If you grab the latest snap shot of the development version of GWindows, there is a new tool (GBManager) in it that allows you to view all the registered libraries / COM objects on your system and bind to any one of them with a few clicks. You will find that even on a clean install of 2K or XP there are hundreds on your system.

References to Publications

COTS Journal - Space Station Robot Embeds Ada

From: Marc A. Criley
*   <mcqada95@earthlink.net>*
Date: Tue, 21 May 2002
Organization: Quadrus Corporation
Subject: Ada Robotica
Newsgroups: comp.lang.ada

The March 2002 issue of "COTS Journal" contains an article on the use of Ada on the Space Station's Canadarm2. The issue's index is at
http://www.rtcgroup.com/cotsjournal/2002/03/
scroll down to the "COTS View" section. [Cf. also "Canada - Space Station Robot Embeds Ada" further in this news section. -- dc]

Marc A. Criley, Consultant, Quadrus Corporation, www.quaduscorp.com

DDC-I Online News

[Extracts from the table of contents. See elsewhere in this news section for selected items. -- dc]

From: JC <jcdk@ddci.com>
Date: Thu, 30 May 2002
Subject: Real-Time Industry Updates - News from DDC-I
To: F9DK May 2002 Online News <jcdk@ddci.com>

DDC-I Online News, May 2002, Volume 3, Number 5

A monthly news update dedicated to DDC-I customers & registered subscribers.

Three Year Anniversary for the Danish Micro Satellite "Oersted" DDC-I Ada Compiler System (DACS) Onboard... Yet Another Proven Success!

Third Party Update: TopGraph'X. Offering middleware to reduce the cost and complexity of embedded systems development.

For the complete newsletter, go to:
http://www.ddci.com/news_vol3num5.shtml

From: JC <jcdk@ddci.com>
Date: Thu, 27 Jun 2002
Subject: Real-Time Industry Updates - News from DDC-I
To: G9DK June 2002 Online News <jcdk@ddci.com>

DDC-I Online News, June 2002, Volume 3, Number 6

SCORE for OSE. A strong platform for safety critical software development!

Third Party Update: OSE Systems, an Enea Company. A safety certified real-time kernel for today's complex embedded applications.

For the complete newsletter, go to:
http://www.ddci.com/news_vol3num6.shtml

From: JC <jcdk@ddci.com>
Date: Fri, 2 Aug 2002
Subject: Real-Time Industry Updates - News from DDC-I
To: H9DK Aug 2002 Online News <jcdk@ddci.com>
DDC-I Online News, August 2002, Volume 3, Number 7

TADS-1750A Launches the Next Generation Atlas V Rocket. Lockheed Martin depends on TADS to develop flight control software.
DACS-MAPP.

Multi-application programming with paging. New from DDC-I.

Ada95 Training.

Third Party Update: Vector Software. The next generation of intelligent test tools!

For the complete newsletter, go to:
http://www.ddci.com/news_vol3num7.shtml

Ada Distilled - On-line Book Updated

From: Richard Riehle <richard@adaworks.com>
Date: Sat, 01 Jun 2002
Organization: AdaWorks Software Engineering
Subject: Announce: Ada Distilled Update
Newsgroups: comp.lang.ada

There is an updated version of Ada Distilled in PDF format available for free download at
http://www.adaic.org
http://www.adapower.com

It is my intention to continue to refine and update this little booklet on Ada for as long as there is continued interest. My thanks to the many users who have sent me suggestions and thanks over the past year. I am trying to keep it short and simple so it can be used by any programmer experienced in some other language as a quick start into the language.

[A user responded: -- dc]

Thanks, it's a great job, I like it because it's short, simple, but touches all Ada interesting features with lots of examples. Very good for learning what Ada is capable of, and as reference too.

Ada-Europe Conference Proceedings

From: Dirk Craeynest <Dirk.Craeynest@cs.kuleuven.ac.be>
Date: Fri, 7 Jun 2002
Subject: Re: Press Release - Reliable Software Technologies, Ada-Europe'2002
To: ada-france@ada-france.org

[In reply to a question about the availability of the proceedings of the 7th International Conference on Reliable Software Technologies - Ada-Europe'2002. This response is valid for the whole Ada-Europe conference series. -- dc]

Note that the message you are replying to mentioned: "The proceedings published by Springer will be distributed at the conference, and can already be checked out at
http://link.springer.de/link/service/series/0558/tocs/t2361.htm ".

So yes: the proceedings *are* already available. They can be ordered directly from Springer in printed form, and can be consulted on-line on their web-site (this requires a s.c. "LINK account").

And of course, the proceedings will be handed out to each participant as has been common practice at the Ada-Europe conferences. Finally, the registration form enables participants to order extra copies of the proceedings at a special reduced price. [...]

[Further clarification: -- dc]

You can browse the table of contents and the abstract of each paper in the proceedings on-line. The full text of each paper is available as a PDF file, but as said for that you need a (paying) "LINK account" from Springer (as Springer has the copyright). Note that many universities have such accounts for their staff and students.

Dirk Craeynest, Ada-Europe'2002 Publicity Chair

Le Monde Informatique - Ada Overview Article

From: Laurent Pautet <pautet@inf.enst.fr>
Date: Tue, 2 Jul 2002
Subject: [ada-france] article dans le monde informatique du 28 juin
To: ada-france@ada-france.org

For those interested, there is an article on Ada in "Le Monde Informatique". [...]

[The article was entitled "Ada, discret mais costaud" by Gérard Canesi, and was printed in LMI 945 of 28 June 2002. It is no longer available at the
www.weblmi.com
site, but use the title to find the cached version in
www.google.com
-- dc]

On-line GNU Ada Run-Time Book

From: Javier Miranda <jmiranda@iuma.ulpgc.es>
Date: Mon, 15 Jul 2002
Subject: [GNAT-RTS Book] Beta version 0.6 available

[First announced in an e-mail on July 2nd, this is an updated release. -- dc]

The third chapter of the book (Rendezvous) has been re-structured, completed, and upgraded.

Title: A Detailed Description of the GNU Ada Run-Time
Version: Beta 0.6
Last Update: July, 15, 2002
Author: Javier Miranda. Access:
http://www.iuma.ulpgc.es/users/jmiranda

Chapters: 1. The GNAT Project. 2. Task Types and Objects. 3. The Rendezvous. 4. Protected Objects. 5. Interrupts.

Goal: To provide to the Ada community an On-line book which can be used for research and for teaching.

Brief Description: HTML Document which describes the behaviour of the Run-Time of the free Ada compiler (GNAT). Its main features are:

It is linked with the GNAT 3.14p run-time sources. This allows the reader to verify the contents of the book by clicking the html reference to the source file where each concept is handled by the Run-Time.

The GNAT source files are also linked. This enables the reader a quick access to the point of declaration of the Run-Time variables and types, to follow the calls to subprograms, etc. and to go backward.

It is linked with the Annotated Ada Reference Manual (Technical Corrigendum 1). Any reference to the AARM has a link to the corresponding section. Most of the papers referenced in the book are also included in PostScript and PDF formats.

Javier Miranda, University of Las Palmas de Gran Canaria, Canary Islands, Spain

The Big Online Book of Linux Ada Programming - Update

From: PegaSoft Canada - ALT Drop Box
  <adalin-l@tiamet.vaxxine.com>
Date: Sun, 14 Jul 2002
Subject: Big Online Book Update - C/C++
To: GNAT Discussion List
  <gnatlist@lyris.seas.gwu.edu>

[Cf. also the same topic in AUJ 21-3 (October 2000), p.171. -- dc]

I've updated the Big Online Book of Linux Ada Programming. Chapter 19 now includes sections on mixing Ada with C and C++. Please let me know if there are any corrections that should be made. If anybody wants to submit their experiences with GCC 3.1 and GNAT 5.0, I'll add it to the book. The book is located at
http://www.pegasoft.ca/homes/book.html

Ken O. Burtch,
http://www.pegasoft.ca
ken@tiamet.vaxxine.com
Pegasoft, Jordan Station, ON, Canada

CrossTalk - Keynote STC Speech Lloyd K. Mosemann

From: Nielson Mark S Civ OO-ALC/TISEB
  <Mark.Nielson@hill.af.mil>
Date: Fri, 19 Jul 2002
Subject: The August 2002 Issue of
  CrossTalk is now available on-line.
To: <Dirk@offis.be>

The August 2002 issue of CrossTalk, The Journal of Defense Software Engineering is now available on our Web site at:
http://www.stsc.hill.af.mil

[...] If you missed the 2002 Software Technology Conference, you will not want to miss reading Lloyd K. Mosemann's "Did We Lose Our Religion?" This reprint of his speech at the April conference in Salt Lake City takes a hard-line look at the progress the government has made and not made in predicting software cost, quality and on-time delivery. [...]

A naming convention for classes in Ada

From: Jean-Pierre Rosen
  <rosen@adalog.fr>
Date: Thu, 25 Jul 2002
Organization: Adalog
Subject: Re: Naming conventions
Newsgroups: comp.lang.ada

Coming from C++, what I used to think of as a class (a type and its associated methods) is now roughly equivalent to a package. Whenever I try to name a package, I use what I normally think of as the type itself. For example, I would put a vector type in a package Vector or a regular expression type in a package Regular_Expression. However, once I name the package, I am at a loss for what to call the data type inside.

See "A naming convention for classes in Ada 9X", Ada Letters Vol XV n° 2, for a discussion of a solution to this problem. You can also download the paper from:
http://www.adalog.fr/publica2.htm

Linux Journal - Testing Safety-Critical Software with AdaTEST

From: Tucker Taft
  <newsreader@yaletaft.com>
Date: Sun, 28 Jul 2002
Organization: AverStar
Subject: Ada article in Linux Journal
Newsgroups: comp.lang.ada

There is a nice article about the "AdaTEST" product in the embedded Linux part of the online Linux Journal. [...] The URL for it is
http://www.linuxjournal.com/article.php?sid=5965

Ada Inside

Denmark - Three Year Anniversary for the Danish Micro Satellite "Oersted"

URL:
  http://www.ddci.com/news_vol3num5.shtml
Subject: DDC-I Online News, May 2002,
  Vol. 3 Issue 5

Three Year Anniversary for the Danish Micro Satellite "Oersted". DDC-I Ada Compiler System (DACS) Onboard. By Erik Jensen, President DDC-I A/S

[Cf. also "Commercial satellite" in AUJ 20-1 (April 1999), pp.41-42. -- dc]

On February 23, 2002 the first Danish Micro Satellite celebrated its 3rd birthday in orbit, transmitting information for 2 years longer than originally expected. The satellite continues to provide interesting data concerning the earth's magnetic field.

[...] The satellite was placed in orbit 850 kilometers above the earth. Every day it circles around the earth 6 to 8 times, and transmit its data about the earth's magnetic field and particle radiation from the sun. The information is gathered by three different antennae located in Denmark on two universities and on the Danish Meteorological Institute in Lyngby close to DDC-I headquarters in the same suburb of Copenhagen.

In November last year, some of the scientific results were published by the international scientific magazine "Geophysical Research Letters" by a team of 27 scientists headed by Nils Olsen from the Danish Space Agency.

Ideas about making the first Danish micro satellite date back to the early 90's, and initial funding was established in 1993. DDC-I became involved with the project early on and the software for the satellite was developed with the DDC-I Ada Compiler System (DACS) by engineers from several players in the Danish Industry including TERMA & DDC-I. Many other scientific organizations and universities have been involved with the program including the Danish Space Agency, DMI, DTU, Aalborg University, ESA, NASA and many more.

Additional information is available at
www.oersted.dk
(currently only in Danish). [...]

Canada - Space Station Robot Embeds Ada

URL:
  http://www.rtcgroup.com/cotsjournal/2002/03/
Subject: COTS Journal Online, March
  2002, Volume 4 Number 3

[...] COTS View: "Case Study: Space Station Robot Embeds Ada",

Robert Dewar, Ada Core Technologies

"Despite rumors of its demise, Ada is at the heart of the International Space Station's in-orbit Canadarm 2 where it assures software safety and reliability." [...]

USA - Digital Maneuvering Boards in Ada

From: Richard Riehle
  <richard@adaworks.com>
Date: Fri, 10 May 2002
Organization: AdaWorks Software
  Engineering
Subject: Digital MoBoard in Ada
Newsgroups: comp.lang.ada

At last year's SIGAda Conference, two of my students presented a paper on a project they did for their Master's Thesis in our Software Engineering program at Naval Postgraduate School. Since their graduation, they have continued to develop and enhance this system. It is

getting a lot of interest within many U.S. Navy groups.

They have recently put up a web site that might be of interest to some of the readers of this forum. It is an example of how Ada along with GtkAda can be used successfully to develop a really useful application. If you like it, drop them a note of encouragement. The address of the site is: www.DigitalMoboard.hssites.com

From: David C. Hoos, Sr.
*   <david.c.hoos.sr@ada95.com>*
Date: Sun, 12 May 2002
Subject: Re: Digital MoBoard in Ada
Newsgroups: comp.lang.ada

Moboards are so-called Maneuvering Boards which are used to manually plot the relative motion of ships approaching each other. There is a nice Power Point presentation used in Naval Officer training classes at http://www.colorado.edu/NROTC/NAVR_Classes/3020_-Nav_II/Rel.Motion-_Moboard_1_.ppt

As with many very powerful and useful computer programs, the need for or the desirability of a particular program cannot even be imagined by someone unfamiliar with the problem domain. My own work is in an area of endeavor that I didn't even imagine existed, until called upon to write software for it.

From: Richard Riehle
*   <richard@adaworks.com>*
Date: Sun, 12 May 2002
Organization: AdaWorks Software
*   Engineering*
Subject: Re: Digital MoBoard in Ada
Newsgroups: comp.lang.ada

[...], the two Naval officers doing the development are continuing to expand the capabilities of their software. If the Navy eventually adopts it on a larger scale, it will be an excellent Ada project. [...]

[From another message: -- dc]

At this stage of the project, it is no longer a student project. It has moved on to become an important piece of software that the Navy might be able to incorporate into more significant shipboard systems. Already, one trial version is deployed using a laptop computer on one ship. [...] The software is being updated to interface with some serious data acquisition systems that will be in the realm of classified software.

[...] My only reason for calling attention to the site was to let people know that there was an active project in Ada that had some promise. [...]

USA - COTS Ada Software Package for Avionics Applications

URL: http://www.adaic.org/news/act-wind.html
Subject: Ada Info Press Release--Ada Core Technologies and Wind River Team with

Smiths Aerospace in Safety Critical Ada Development

Ada Core Technologies Teams for Embedded Ada development

Ada Core Technologies, Inc., and Wind River Systems, Inc., are working with Smiths Aerospace to create a software package that both complies with strict avionics software standards and yet is also a commercial off-the-shelf (COTS) software solution for safety and mission-critical embedded systems. Smith Aerospace, which develops software applications and integrating systems that use software partitioned architectures, will fund, market, and manage the project. The company is integrating several avionics software applications, including the Smiths Flight Management System (FMS) and the Communication Management function, onto a single PowerPC processor for the C-130 AMP and 767 Tanker Transport programs.

The PowerPC processor will host Wind River's Tornado for Safety Critical Systems platform. Tornado is part of Wind River's integrated embedded software solutions for connected devices. The platform supports the development of a wide range of embedded and desktop systems, from aerospace avionics applications to automotive and telecommunication devices.

Ada Core Technologies (ACT) is providing support with its GNAT Pro environment for Ada software development. [...] For more information on the new team for embedded Ada development, please see ACT's press release.

["ACT - Wind River Partners with Smiths Aerospace and Ada Core Technologies to Deliver Tornado for Safety Critical Systems" earlier in this news section. -- dc]

USA - ObjectAda/Raven for JSF Jet Engine

URL:
*   http://www.aonix.com/content/news/pr_6.14.02.htm l*
Press Release: Pratt & Whitney has Selected Aonix ObjectAda/Raven For its Next Generation Military Jet Engine for JSF

Boulder, Colorado, June 14, 2002 Aonix, a member of the Gores Technology Group and a leading provider of Ada 95 software development environments and safety critical solutions, has announced that Pratt & Whitney has selected ObjectAda/Raven for its next generation military jet engine program. Based on the successful completion of the commercial PW6000 engine program, Pratt & Whitney will use the Aonix ObjectAda/Raven product line on the JSF F-35 program for their F135-PW-100 jet engines.

[Cf. also "USA / Pratt & Whitney - Commercial Jet Engine Certified With Aonix ObjectAda/Raven" in AUJ 23-1 (March 2002), pp.35-36. -- dc]

Pratt & Whitney originally selected the ObjectAda/Raven product for use on the PW6000, commercial jet engine program, in 1998 following an extensive evaluation. [...] The Pratt & Whitney certification was achieved at software Level-A of RTCA's DO-178B, Software Considerations in Airborne Systems and Equipment Certification.

Certification to DO-178B requires that all COTS software included in the product be certified to the same standard as the core product. ObjectAda/Raven meets this standard for Level-A systems. Pratt & Whitney incorporated the full suite of lifecycle artefacts provided by Aonix within the overall engine-system software documentation set approved for certification by the Federal Aviation Administration.

[...] Aonix is the leading supplier of certified (i.e., certifiable) run-time systems for the Ada 83 language and offers the only certified Ada 95 run-time system. ObjectAda/Raven is certified to the highest certification levels and meets the DO-178B Level-A required by the FAA for airborne systems.

[Cf. announcements of Aonix products earlier in this news section for general information on Aonix and contacts. -- dc]

USA - Green Hills for New Sikorsky S-92 Helicopter

URL: http://www.ghs.com/news/220801r.html
Rockwell Collins Selects Green Hills Software's INTEGRITY-178B(TM) RTOS To Fly In New Sikorsky S-92 Helicopter

Santa Barbara, CA. August 1, 2002 -- Green Hills Software today announced that Rockwell Collins has integrated the INTEGRITY-178B(TM) real-time operating system (RTOS) into the Sikorsky S-92 helicopter's new highly-integrated avionics package. INTEGRITY-178B, along with Green Hills Software's GSTART Ada run-time environment, is being used in Rockwell Collins new Avionics Management and Display System, which incorporates the most advanced avionics technology available today for rotorcraft. [...] The new S-92 cockpit is designed for outstanding visibility and is equipped with a highly-integrated avionics package which provides the core of an open architecture avionics suite for processing aircraft system information. Flight data is shown on four Collins multi-function displays, with a fifth display offered as an option.

Green Hills also announced that Rockwell Collins is using Green Hills Software's

AdaMULTI Integrated Development Environment (IDE) to develop the flight software for the new S-92 cockpit.

The S-92 is Sikorsky's newest medium-lift helicopter and is designed to meet both civil and military requirements. Featuring a passenger capacity of 19-22, the versatile new helicopter will serve a variety of commercial and international utility needs, including passenger, cargo, aeromedical, search and rescue and resource development support.

A Motorola PowerPC running INTEGRITY-178B and the GSTART Ada run-time environment provides the backbone for the Sikorsky helicopter's new Collins Avionics Management and Display system. The system provides the display and integrated management of primary flight data, presentation and management of navigation information for the S-92. The system also provides flight management data, a digital map, weather radar, terrain information and engine instrument caution and advisory system processin and display. Rockwell Collins will use INTEGRITY-178B and GSTART to achieve DO-178B Level A certification of the Avionics Management and Display system.

"INTEGRITY-178B is the RTOS of choice for safety-critical applications like the S-92 helicopter, where the overwhelming emphasis is on safety, security and high integrity software development," said John Carbone, vice president of marketing for Green Hills Software.

"INTEGRITY-178B's robustly partitioned architecture, real-time deterministic performance, and overall ease-of-use of the AdaMULTI tool chain gives it a clear advantage over general purpose RTOSes like VxWorks."

[...] INTEGRITY-178B includes an RTOS simulator (ISIM) that enables programmers to develop and test their code on a PC or workstation without the need for target hardware. INTEGRITY-178B also features a real-time event analyser (EventAnalyzer(TM)) that enables viewing of system and user events in a graphical display.

INTEGRITY-178B is tightly integrated with Green Hills Software's AdaMULTI 2000 IDE. Together with the Green Hills family of optimising Ada 95, C, and C++ compilers, AdaMULTI automates all aspects of embedded software development, including editing, source-level debugging, program building, run-time error checking, version control, and code/performance optimization. AdaMULTI also features an advanced code coverage tool (GCOVER(TM)) that automates structural coverage analysis for application software, [...]

About Green Hills Software, Inc. [Cf. announcement from IPL earlier in this news section for general information on Green Hills Software. -- dc]

For More Information Contact: Green Hills Software
Lynn J. Robinson
lynnr@ghs.com
Davis-Marrin Communications
Michelle Ragsdale
michelle@davismarrin.com

USA - Next Generation Atlas V Rocket

URL:
  http://www.ddci.com/news_vol3num7.shtml
Subject: DDC-I Online News, August 2002, Vol. 3 Issue 7

TADS-1750A Launches the Next Generation Atlas V Rocket.

Lockheed Martin Space Systems depends on TADS to develop modern, object-oriented real-time flight control software for the new Atlas V launch vehicle.

The maiden voyage of Lockheed Martins first Atlas V rocket is scheduled to liftoff from Space Launch Complex 41 at Cape Canaveral on August 12, 2002, freed from its earthly bonds by flight control software created with DDC-I's Tartan Ada Development System (TADS).

Incorporating numerous systems developed for the flight-proven Atlas III, development of the Atlas V the most powerful Atlas launch vehicle ever built began in 1998, to meet the growing needs of the U.S. Air Force Evolved Expendable Launch Vehicle (EELV) program and International Launch Services (ILS) commercial and government satellite customers worldwide.

"Every Atlas variant during the past ten years has had a flawless first flight," says ILS President Mark Albrecht. "With a modular design and increased capability, the Atlas team provides a vehicle that meets a wide range of customer needs. To the development team, I say thank you."

According to Michael Bethancourt, Flight Software Development Lab Manager and Team Leader for the Atlas Flight Software Operating System (OS) group, [...] "We have a highly experienced team, and what we did, moving from Atlas II to Atlas III, was to restructure and layer the OS in a more object-oriented fashion. This affected much of the Atlas III software" he explains. "When we started on Atlas V, what we did was bring forward our sophisticated and proven control algorithms and implement them into a software structure that was fully object-oriented. We employ a number of powerful Ada95 features, not all of them, but we essentially have Ada95 here."

He adds that Ada is the appropriate language for much of what they do. The modern features that are critical to building reliable and robust flight software, including strong typing, dynamic binding, polymorphism, and a secure environment merge to produce far fewer interface errors, and an effortless enforcement of proven development standards. "Ada is definitely the right language for mission critical, safety critical, real-time embedded systems" he adds.

Describing the stability and performance of the AdaTest95 processor and the file-coupled TADS system as "turnkey," he adds that the multi-window interactive symbolic debugger within the 1750A simulator and target chipset is a major advance over previous development subsuites.

"We have a homegrown debugger built into our flight box downstairs for Atlas II & III. We did full module testing on our test set downstairs, using capabilities written years ago on a DEC VAX. We still do full module testing for II & III on the VAX and we go down and do our branch testing on that flight-equivalent test set."

The branch testing combined with the Computer Software Component and system testing where the software is "flown" against a simulation of the vehicle provides the team with solid confidence in the reliability and robustness of their software.

"With the Atlas V, instead of having to do all of our branch testing on the single test set, now we can run against any one of a number of single-board computers with the debugger in place. We can sit up here at our desks and test on computers that are actually located in our Software Integration Laboratory that is nearly one mile away. We get much more accomplished, and the whole process is just more productive."

With 34 years in software, including a stint teaching OS internals, and 26 years of hands-on experience in the real-time embedded systems world, Bethancourt is well-equipped to comment on the state of programming tools [...]

Finland - On-Board Software for GOCE Satellite & Static Analysis Tool

From: Niklas Holsti <holsti@ssf.fi>
Date: Thu, 1 Aug 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

One data point from Europe: at Space Systems Finland, a small private company specialising in space software (both space- and ground-based), we have two current Ada projects, which you may include in your list with attribution.

The on-board Platform Application Software for the GOCE satellite (Gravity and Ocean Circulation Explorer), an ESA

satellite which will run in low polar orbit and use a sensitive gravity gradiometer to measure the Earth's gravitational field. The Platform Application Software runs on the central processor (ERC32 = SPARC V7) and includes the data handling (TC/TM) functions and the attitude and orbit control functions, with ion engines. Software development started in January 2002 and will use the XGC Ada compiler (a variant of Gnat) with the Ravenscar tasking profile. In this project, Ada was requested by the customer and we were happy to comply.

A software development tool to compute upper bounds on worst-case execution time, using static analysis of machine code. This tool has been under development since about mid-1999, so it is not a new start. In this project, we were free to choose the implementation language. We chose Ada partly for technical reasons (including portability to Solaris/Linux/Windows) and partly to keep our staff in training for space projects. Of course I would make the same choice again. [...]

[From another message, on the XGC Ada compiler: -- dc]

As I understand it, XGC provide their own kernel (limited to Ravenscar) and have tailored the Gnat front-end to interface more directly with this kernel, without all the layers that are present in the standard Gnat. They also support a number of other profiles that are even less permissive, for more critical systems. Their compiler also excludes certain other Ada language features so that they can make do without the "secondary stack".

We chose the XGC compiler and kernel for this project based on a comparison of the real-time performance between various Ada systems for the ERC32 processor. This comparison was made by ESA/ESTEC some time ago, [...].

XGC sells a complete package tailored to the ERC32, including start-up modules etc., with technical support. Our project requirements originally included some tasking features (CPU time and real-time monitoring) that would have required kernel extensions, [...]

These requirements have later been reduced, so we can now make do with just the standard Ada features (Ravenscar subset, of course). [...]

Niklas Holsti, Space Systems Finland Ltd., Espoo, Finland
Niklas.Holsti@ssf.fi

List of Real-World Projects Powered by Ada

From: Michael Feldman
  <mfeldman@seas.gwu.edu>
Date: Mon, 29 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

[...] Obviously Ada will be around in "legacy" (sub-)systems for a long time to come. [...] Is anyone aware of new starts [...]? As you probably know, I try to keep abreast of this, and report it in
http://www.seas.gwu.edu/~mfeldman/ada-project-summary.html [...]

From: Michael Feldman
  <mfeldman@seas.gwu.edu>
Date: Thu, 1 Aug 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

Well, maintaining that list is only a small hobby for me, among many other job responsibilities and hobbies. Most of the items in that list came from tips I got from Teamers. As advocates, we all seem to keep our antennas up looking for any hints of Ada out there, [...]

There's only so much "staying in touch" I have time for - I need to depend on other advocates (like y'all out there). As for "freedom to report" - as you all know, I (like any good "investigative journalist") protect my sources.

Certainly, if a project is mentioned in a sponsor's or implementer's website, it's not confidential, so nobody needs to fear pointing me to the web. Even if nothing is on the web, I will add a listing on the say-so of a Teamer tipster who claims to know.

I know very well that nobody wants to get sacked over revealing the language used in a project. If there are a *lot* of Ada projects for which that is really an issue, I guess it's good news that there are so many projects, but bad news that it'll never help us break out of the Catch 22.

Bottom line: if you know something that will help, use your judgement about whether it's safe to tell me. Write from a 1-shot hotmail account if necessary.:-)

A European Perspective on New Ada Projects

From: Peter Amey <peter.amey@praxis-cs.co.uk>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

Maybe this is just a European perspective, but I don't think things are nearly as black for Ada as the majority of posts have implied. I can think of a dozen or so new-start Ada projects without straining too hard. We were even involved in a new-start Ada 83 project very recently (please don't ask why this strange choice was made!).

Furthermore, I detect a real swing back from the "language doesn't matter, use COTs, hack it together (in a CMM Level 5 hacking shop, of course)" mentality. Some of this is being driven by the security world, especially post 9/11, who under the auspices of the "common

criteria" are obliged to take a more rigorous approach to software integrity. It is also revealing the way my Crosstalk article "Correctness by Construction - Better Can Also Be Cheaper*" was picked up and quoted in the opening keynote address at this year's STC conference by Lloyd K Moseman of SAIC: he made a particular point about the need to restore engineering discipline to the software world. I have also had a terrific response to that article, from people who have had enough of the process dominated approaches of recent years.

In this environment, it is essential that Ada advocates keep fighting and don't let the weaker solutions get chosen by default.

Stress the risk of choosing poorer technologies. Point out that current ways of working don't exactly have a glowing reputation for effectiveness. Tell the pointy-haired managers they might go to jail if they knowingly ignore a proven better technology and then get something wrong. If there is a staff shortage, train people. If people don't think Ada is worth learning then pay good Ada engineers more until it _is_ attractive to learn (this makes sound business sense given the proven quality and productivity increases Ada brings).

Regards, Peter

* Available on
www.sparkada.com

[See also "Crosstalk articles" in AUJ 23.2 (June 2002), p.77, and "CrossTalk - Keynote STC Speech Lloyd K. Mosemann" earlier in this news section. -- dc]

Indirect Information on Ada Usage

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

From: Colin Paul Gloster
  <Colin_Paul_Gloster@acm.org>
Date: Fri, 17 May 2002
Subject: opinion in Space Systems Finland Ltd. as relayed in DASIA 2002
To: team-ada@acm.org

In this year's DAta Systems In Aerospace conference which ended yesterday, one of the very first presentations was entitled "Software Development for a Deep Drilling Micro Rover for Mars Exploration". Therein the speaker described development using Ada 95 targetting Linux. In the questions and answers session, he was asked why they were so conservative in choosing Ada as the language but so adventurous in choosing Linux as the operating system. Whatever about Linux, Ada was chosen because it is safe and reliable.

*From: postman@jobscareer.be
  <postman@jobscareer.be>
Date: Fri, 17 May 2002
Subject: Your weekly jobscareer.be postman*

Unix software developer (Ada and C++),
Oost-Vlaanderen, Belgium.

[...] Design, development, modification
and testing of software. [...] Experience in
OO programming. Knowledge of Ada is a
must [...]

*From: postman@jobscareer.be
  <postman@jobscareer.be>
Date: Fri, 24 May 2002
Subject: Your weekly jobscareer.be postman*

* Ada Software Engineers, All of
Belgium [...] You will be integrated in a
strategic project including architectural &
detailed design, development of the
application, programming, testing and
writing of the design documentation.
Knowledge of a programming language
preferably Ada 83-95 or C++ [...]

* Ada Developer-Tester, Brussels [...] For
a key partner in the business activity of
real-time software, integration & testing,
this company needs experienced Ada
analyst programmers to work on a high-
risk Transport project; these Ada
specialists will be actively involved in
different phases of the project, going from
analysis to development & testing. The
Ada technicians we are looking for should
be strong in analytical work, in
communication, teamwork and should
also be open to international experiences.
We have opportunities for some senior
Ada specialists to lead the
developers/testers team. [...]

* Ada Tester Analyst, Brussels [...]

*From: David Rasmussen
  <pinkfloydhomer@yahoo.com>
Date: Fri, 31 May 2002
Subject: Localized Variable Declaration
Newsgroups: comp.lang.ada*

I am a C++ programmer learning Ada. So
far, I love it. It is more or less the answer
to my prayers. [...] really enhances
readability and minimizes bugs (two
things that Ada excels at), [...]

*From: PlanetRecruit.com
  <mailout@planetrecruit.com>
Date: Tue, 04 Jun 2002
Subject: ** 2 NEW Jobs from
  PlanetRecruit.com (04/06/2002) ***

Ada Software developer (Ada 95),
Brussels, Belgium

Client req's an Ada 95 developer who has
at least 3 years experience of development
on UNIX. Knowledge of OOAD
techniques necessary (HOOD, BOOCH or
UML/Rational Rose), C++, and ATC is
an advantage. Need to be team spirited
and deadline focussed. Will be involved
in whole life-cycle of SW application dev.
[...]

Date: Tue, 11 Jun 2002

*Subject: BE-Vlaams-Brabant-Software
  engineer Ada*

Software engineer Ada, Vlaams-
Brabant/Oost-Vlaanderen, Belgium,
FullTime, Permanent

Company specialized in IT projects with a
high added value covering following
domains: Application Integration &
Support, Real-Time Embedded Systems,
Software Quality Management,

[...] The more experienced applicants will
be responsible for Object Oriented
Analysis review, the architecture and
design of applications, [...] The missions
will also involve analysis, software
design, and documentation and testing.
The consultants will be involved in the
validation of the design, the performances
and the reliability.

[...] Profile: Several years of experience in
the specification and design of application
software, preferably in a critical
environment. [...] Sound experience in the
production of technical specifications for
systems. Very good knowledge of the
programming language Ada. Good
knowledge of Unix and NT Platforms.
Good knowledge of CORBA and/or UML
and/or Oracle are advantages.

*From: CONEX Consulting AG
  <info@conexconsulting.com>
Date: Thu, 13 Jun 2002
Subject: JOB-Oportunity for Ada
  Programers*

We are looking for several programmers
with knowledge of Ada. [...] Skills: Ada,
Motif, VMS, CMS, Databases Sybase,
Oracle. [...] City: Financial institution in
Bern/Switzerland [...]

*Date: Wed, 26 Jun 2002
Subject: Ada in Brussels (Walloon Brabant)
To: dirk.craeynest@cs.kuleuven.ac.be*

Ada S/W Developer, Brussels (Walloon
Brabant), Belgium, Permanent

3 x Ada S/W Developers (junior or
expert) are required for a rapidly growing
global organisation. Candidates will be
responsible to develop an application,
from the users requirements to the
customer acceptance, following the
complete project life-cycle. Technical
experience should include an Ada S/W
development background (1-7 years).

*From: PlanetRecruit.com
  <mailout@planetrecruit.com>
Date: Wed, 03 Jul 2002
Subject: ** 1 NEW Jobs from
  PlanetRecruit.com (03/07/2002) ***

Ada Software Developers

This fantastic organisation is looking for
Ada developers to join their Belgian
operation in Brussels on a permanent
basis. Candidates should be fluent in
English and/or French and be comfortable
talking through technical requirements in
these languages. Skills of interest include:
Ada minimum 2-3 years experience.

Knowledge of Unix would is essential, as
well as UML and OO design. [...]

*From: Arne Carlsson
  <arne.carlsson@space.se>
Date: Fri, 05 Jul 2002
Organization: Saab Ericsson Space
Subject: GNAT on SPARC
Newsgroups: comp.lang.ada*

I am working with GNAT (in fact GNAT-
ORK) for an embedded SPARC sytem
(ERC32 processor). [...]

*From: postman@jobscareer.be
  <postman@jobscareer.be>
Date: Fri, 05 Jul 2002
Subject: Your weekly jobscareer.be postman*

Informaticiens Industriels Maitrisant Ada,
[...]

Ada n'a plus de secret pour vous? Grand
groupe industriel recherché pour son
entité de Charleroi des informaticiens
expérimentés en informatique industrielle
pour diverses missions nationales et
internationales. Connaissances
nécessaires en langages assembleurs x86,
C, C++, Java ainsi que Ada. Maîtrise du
Français et de l'Anglais et bonne
connaissance du monde industriel. [...]

*From: Master Consulting
  <gaspard.schetz@masters-
  consulting.net>
Date: Mon, 8 Jul 2002
Subject: cherche des bons programmeurs
  Ada pour la Belgique
Newsgroups: fr.comp.lang.ada*

Pourquoi ne pas changer d'horizon? Et
venir en Belgique... On y mange bien, les
femmes sont jolies, c'est un peu la France
tout en étant à l'étranger, [...] Je suis prêt à
entendre vos rêves en Ada....

*From: Tarroux
  <philippe.tarroux@limsi.fr>
Date: Wed, 17 Jul 2002
Subject: RE: [ada-france] tash
To: <ada-france@ada-france.org>*

J'utilise tash pour développer des
interfaces pour nos outils de recherche
dans la mesure où nous avons à la fois des
machines Unix, des PCs et quelques macs
aussi. [...] Le fait de pouvoir utiliser avec
Ada la dernière version de Tcl (8.3.4.2)
qui est pleine de trucs nouveaux
(nouveaux widgets et nouvelles options,
interface sons (Snack)...) est assez sympa.
[...]

Philippe Tarroux, Groupe Perception
Située, LIMSI-CNRS, Orsay, France

*From: Davenport, Darren
  <ddavenpo@ball.com>
Date: Mon, 29 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org*

Here at Ball Aerospace, all commercial
birds are going to be in Ada indefinitely.
So far: GFO, QuikSCAT, MTI,
Quickbird, soon to be launched ICESat,
and just won NPP (NPOESS Preparatory
Project). We don't have trouble finding

software engineers but we're in Boulder, CO which may help.

Darren Davenport, Ball Aerospace and Technologies Corp, Boulder, CO

From: Tony Lowe <tlowe@issintl.com>
Date: Mon, 29 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

[...] P.S. A friend who works for a big Sat-Com firm says they are highly considering Ada for their future projects...

From: Hart, Hal <Hal.Hart@trw.com>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

[...] we [TRW] have a sister division in Huntsville and avionics business elsewhere (in a somewhat organizationally remote group called Space & Electronics) from which I have heard reports in the last year of proposals being done in Ada.

From: Emil Moholth
  <emil.moholth@kongsberg.com>
Date: Thu, 8 Aug 2002
Subject: Re: GNAT in GCC 3.1 breaks
  legacy code
Newsgroups: comp.lang.ada

Interesting, how through CORBA, you can use all the languages you like!

And across computer architectures. We're running Ada on PPC with vxWorks, and Ada, Java and C++ on Solaris, all communicating through CORBA.

From: Bonnin Hugues
  <hugues.bonnin@thales-avionics.com>
Date: Tue, 20 Aug 2002
Organization: Thales-avionics
Subject: Re: pb. with big object code
To: GNAT Discussion List
  <gnatlist@lyris.seas.gwu.edu>

I have a very big Ada unit to compile (one Library Unit in 2500 files, and 400 KLOC), and the compiler (GNAT 3.13p or 3.14p under NT) has problems to generate the object [...]

Ada Embedded Community Surveyed

URL: http://www.adaic.org/news/survey.html
Subject: Ada Info Press Release--Ada in
  Embedded Systems 2001-2002

Ada Embedded Community Surveyed

[Cf. also "Ada Survey" in AUJ 23-2 (June 2002), p.79. -- dc] In a 2002 survey by the independent market research firm VDC, embedded Ada developers named the language's ability to find bugs in compile time as its major benefit. Its nature as an ISO standard and object oriented language were Ada's next important advantages. Almost half of those surveyed believe that Ada reduces a project's costs. VDC's results are available in the .pdf file Ada in Embedded Systems (five pages).

[Cf. http://www.adaic.org/news/survey.pdf -- dc]

Ada in Context

Avoiding Traps and Pitfalls

From: Richard Riehle
  <richard@adaworks.com>
Date: Thu, 06 Jun 2002
Organization: AdaWorks Software
  Engineering
Subject: Re: Embedded Ada Development
  Tools
Newsgroups: comp.lang.ada

I just took a quick tour through a new book from Microsoft Press titled, Writing Secure Code. It would be hilarious if it were not so pathetic. Over and over the authors demonstrate how to avoid the pitfalls of writing code in C and C++. Over and over I find myself reacting with, "This problem simply would not occur if they were writing in Ada."

The authors are well-intentioned and intelligent. They even admit, early in the book, that they are writing about C code because that is the more widely used language even though it is clearly less secure and less safe than many other languages. Of course, they never mention Ada since they probably know nothing about it.

One of my favorite monthly columns in C++ Report was called "Obfuscated C++." It was also a source of amusement because the obfuscations described simply could not have happened in Ada. Still, these "professionals" continue to sell their souls to the Devil when it would be so much easier to harmonize with the angels.

From: Wes Groleau
  <wesgroleau@despammed.com>
Date: Tue, 18 Jun 2002
Organization: Raytheon Company
Subject: Re: Faulty languages and Liability
Newsgroups: comp.lang.ada

Does anybody know of a list of typical C/C++ errors that can't happen in Ada? It might be useful to do a side-by-side comparison.

I read the first half of "C Traps and Pitfalls" Almost all of them either could not happen in Ada or would only happen if you knowingly forced them to. The best one was more than a page discussing the subtle differences between various long strings consisting almost entirely of asterisks and parens!!

Why Choosing Ada? - Some Resources

From: Hugues Mulliez <hmu@sogetek.fr>
Date: Tue, 2 Jul 2002
Subject: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

[Thread translated from French: -- dc]

I'm looking for a site explaining why to choose Ada rather than another language: which advantages can one draw from this language compared to C++ for example. What are the comparative data with other languages, that would justify me to choose Ada for a given project. This with the aim of making a presentation of the language Ada, short, general, for the non-initiated.

From: Simon Claude
  <claude.simon@equipement.gouv.fr>
Date: Tue, 02 Jul 2002
Organization: S.E.T.R.A.
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

See for example
http://libre.act-europe.fr/Software_Matters/

From: Jean-Pierre Rosen
  <rosen@adalog.fr>
Date: Tue, 2 Jul 2002
Organization: Adalog
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

And of course
http://www.adalog.fr
☺

But also the reference site in English:
http://www.adapower.com

There is an "ammunitions" page on the
www.adahome.com
site as well. Pay attention: this site is no longer maintained and contains obsolete information; nevertheless, the basic arguments are still valuable.

From: Gautier
  <gautier_niouzes@hotmail.com>
Date: Tue, 02 Jul 2002
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

Another: http://www.adaic.org/ [...]

From: erreur <dev@newdeal.ch>
Date: Tue, 2 Jul 2002
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

Look in the other newsgroups, the problems they have... and you'll understand why to choose Ada... ☺

From: David Marceau
  <davidmarceau@sympatico.ca>
Date: Sat, 06 Jul 2002
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

The following articles in English should interest you.
http://www.stsc.hill.af.mil/CrossTalk/2002/mar/mar02ind.asp

Page 24: "Correctness by construction". This gives compelling evidence showing that focus on bug prevention rather than bug detection raises quality and saves time and money. This approach was done using Spark (an Ada subset) and full Ada. March 2002 "COTS journal"

http://www.rtcgroup.com/cotsjournal/2002/03/cots03-cotsview.pdf

This gives a list of factors making Ada increasingly popular.

From: Jean-Marc Saffroy
<saffroy@ri.silicomp.fr>
Date: Sat, 6 Jul 2002 11:44:18 +0200
Subject: Re: Aide : pourquoi choisir Ada?
Newsgroups: fr.comp.lang.ada

This amusing page explains how to write code which is impossible to maintain:

http://www.web-hits.org/txt/codingunmaintainable.html

You can read there: "Avoid Ada. About 20% of these techniques can't be used in Ada. Refuse to use Ada. If your manager presses you, insist that no-one else uses it, and point out that it doesn't work with your large suite of tools like lint and plummer that work around C's failings." ☺

Jean-Marc Saffroy, Research Engineer, Silicomp Research Institute

On Java

From: Georg Bauhaus <sb463ba@l1-hrz.uni-duisburg.de>
Date: Tue, 6 Aug 2002
Subject: Grace: not alone
Newsgroups: comp.lang.ada

An article by Elliotte Rusty Harold, "10 Reasons We Need Java 3.0", http://www.onjava.com/lpt/a/2524 adds some interesting views to the discussions about Ada not having a decent data structure library. "

Java's current collections API is a hodgepodge of different designs implemented at different times. Some classes are thread-safe (Hashtable, Vector). Some aren't (LinkedList, HashMap). Some collections return null when a missing element is requested. Others throw an exception. Let's settle on some standard idioms and metaphors, and design all the classes to fit them, rather the other way around. Probably the easiest way to do this would be to eliminate Vector and Hashtable completely. An ArrayList can do anything a Vector can do and a HashMap can replace a Hashtable."

It also explains that "Java was the first major language to integrate multithreading as a fundamental feature rather than a special purpose add-on library. Thus, it's not surprising that its designers made a few mistakes and missteps in this area. All of these need to be fixed".

[Ole-Hjalmar Kristensen <oleh@vlinux.voxelvision.no> responded: -- dc]

Someone obviously hasn't done their homework. Besides, I would say Java multithreading looks pretty much like an add-on library...

[Anh Vo <anh_vo@udlp.com> wrote: -- dc]

I wish Mr. Harold did his homework throughly before writing something like this. I have spotted a similar claim by Charles L. Perkins and Michael Morrison in "Java 1.1 in 21 Days" - second paragraph page 527. It looks like that bad CEOs are being catched up by others regarding inaccurate claim /cheating... In fact, I did send an email to Mr. Harold reminding him of his inaccurate claim.

[Darren New <dnew@san.rr.com>]

I guess Smalltalk-80 doesn't count as a major language either.

[Robert Dewar <dewar@gnat.com>]

Or OCCAM, or Simula-67, or Algol-68, or PL/1.

[Robert A Duff <bobduff@shell01.TheWorld.com>]

Right. And others. But even if all such languages prior to Java are *not* considered "major", the quote [...] *still* seems kind of silly. Why should a language designer ignore good ideas from prior languages, just because those languages were not "major"?

[Robert Dewar <dewar@gnat.com>]

Of course I agree with this. But to claim Java is a major language and PL/1 was not is just not sustainable. Huge numbers of large organizations wrote major systems using PL/1, and though in retrospect PL/1 was not successful in displacing COBOL, it certainly achieved more significant penetration than java has done so far in real applications. (And by the way, PL/1 is by no means dead, a lot of new software is still written in PL/1 :-)

Faulty Languages and Liability

From: David Botton <David@Botton.com>
Date: Sat, 15 June 2002
Subject: Faulty languages and Liability
Newsgroups: comp.lang.ada

I have been saying for years the day would come that software authors would start to be found liable for their bugs... the time is approaching...

"Researchers on both sides of the Atlantic say most reported security incidents are due to software defects that could easily be fixed."

[Cf. http://zdnetbusiness.cnet.com/investor/news/newsitem/0-9900-1028-20042961-0.html -- dc]

Using Ada is the first step to solving these problems! Now we need to get some spokes people to help these lawyers go after faulty software manufactures for using faulty languages too :-)

From: Robert C. Leif <rleif@rleif.com>
Date: Sat, 15 Jun 2002

Subject: RE: Faulty languages and Liability
Newsgroups: comp.lang.ada

I was hoping to find a lawyer to speak on software product liability at SIGAda 2002. If anyone knows of an attorney, who wishes to become very rich, this would be a golden chance for that attorney. SIGAda has an abundance of expert witnesses. [...]

From: mjsilva697@earthlink.net (Mike Silva)
Date: 19 Jun 2002
Subject: Re: Faulty languages and Liability
Newsgroups: comp.lang.ada

FWIW, I just read in comp.risks (Digest 22.11) that it is estimated that software faults cost the U.S. about $60 billion per year. They actually refer to "inadequate software testing infrastructure" as the problem, but I think they're confusing the real problem (incorrect software) with one component of the solution (software testing). Anyway, $60 billion is getting up to what they in government circles call "real money".

From: S. Ron Oliver <sroliver@csc.calpoly.edu>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

[...] The key point is, Ada, and other sensible technologies (which eliminates all C-class languages, including Java), are going to be around as long as there are responsible, competent Software Engineers, committed to doing high quality work. While it is true the percentage of people in the software industry who fit those criteria might appear to be shrinking, it is just a matter of time before public outrage, if nothing more, will force the issue for many domains. [...]

From: Michael Feldman <mfeldman@seas.gwu.edu>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

What I'm *really* afraid of is what will happen once the semantics of all the convenience application development environments (e.g. MS Visual Basic, Access, .NET, and also Java/J2EE) will shift sufficiently and companies realize that they've built critical applications on a less than sound foundation.

Old cynics like me will probably agree that these companies will just shrug, scrape up some money, and just recode them using whatever tools are popular at that moment. The motto of this industry is "We never have time to do it right, but we always have time to do it over." Always has been, still is.

On Finding Ada Experience and Training

From: Steven Deller <deller@smsail.com>
Date: Thu, 6 Jun 2002
Organization: Smooth Sailing LLC
Subject: RE: Embedded Ada Development Tools
Newsgroups: comp.lang.ada

could you imagine doing embedded Ada development within a small project team, and the Ada developer was to leave the company... Yikes!... I understand Ada is easier to maintain, but you still need someone off the street who is adept at embedded systems programming and Ada coding...

Emenu has over 20 engineers with EXACTLY that expertise – embedded systems and Ada. Each has over 14 years experience. We cover areas from Southern CA to New England, Texas, Arizona, Minnesota, Maryland and more (our engineers are willing to travel). And we all can train a project's own personnel in Ada, embedded systems, and numerous application areas for embedded systems. We do C++ and other languages, but ALL of us would prefer Ada -- it gives us the best way to express our quality engineering approaches. We use OO, OO tools that generate Ada and pay attention to *architecture* as much as implementation. We cover classes to bits and everything in between.

And Emenu is NOT, by far, the only source of excellent expertise in Ada and embedded systems. If you look, you will find...

Steve Deller, Emenu Incorporated,
Steve.Deller@Emenu-Inc.com

From: John R. Strohm <strohm@airmail.net>
Date: Thu, 6 Jun 2002
Subject: Re: Embedded Ada Development Tools
Newsgroups: comp.lang.ada

Every company that has actually TRIED training programmers to use Ada has learned that it is not at all difficult, and that a good programmer very quickly becomes productive in Ada.

When Boeing mandated Ada for the 777, every subcontractor in the country screamed bloody murder. Boeing insisted. None of them wanted to lose out on the biggest contract they would see for a long time, with the biggest airplane manufacturer in the world, that carried the certain knowledge that whoever took this job WOULD get the follow-on work for the NEXT airplane, and the rules would probably be the same next time around. They all toed the line.

They ALL came in on time and in budget. Despite training costs. Despite, in at least one case, having to scrap a bunch of already-completed work and start over from scratch.

Furthermore, when the companies really, seriously, did the metrics collection and looked at the results, they learned that Ada saved them time and money. The best example is the Pratt & Whitney experience, for which Marin David Condic is the knowledgeable individual who was in the trenches at the time.

[Cf. also "On Languages, Productivity and Quality" in AUJ 22-4 (December 2001), pp.222-224. -- dc]

From: Tony Lowe <tlowe@issintl.com>
Date: Mon, 29 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

Along the lines of 'not enough people to do Ada', is this a real concern or a convenient excuse? [...] I could easily get 20 developers with 15-30 years of programming experience each, up and literate in Ada [...]

I guess I am wondering if the 'not enough people' is a self-fulfilling prophesy that companies do not train Ada developers, support Universities to do so, or work towards alternative resources to get people interested in working within the technology, so "no one is available". They want to use other technologies so they use an easy excuse. [...]

From: Stephen D. B. Wolthusen <wolt@igd.fhg.de>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

Meanwhile, we Ada advocates try to find evidence to rebut the old Catch-22: Ada is not being taught much anymore because the faculty and students don't see the jobs out there, and the employers walk away from Ada because they don't see the graduates who are educated to use it.

[...] So yes, getting students to do Ada is the issue. I've been toying on and off with offering a graduate-level Ada / Formal Methods / Security Engineering course at Darmstadt University since just bit**ing about the problem doesn't help [...]

Stephen Wolthusen, Fraunhofer-IGD, Darmstadt, Germany

From: Dale Jr, William <william.dale.jr@lmco.com>
Date: Wed, 31 Jul 2002
Subject: Re: What's Ada's life expectancy?
To: team-ada@acm.org

The approach here is to have more S/W Engineering classes and make better S/W Engineers? I think this is not effective. In all organizations I have worked most software is written, spec'ed, and architected by non-software engineers. They have rarely had more than one intro. class in programming as an undergraduate and carry on with masters and PhDs in other engineering areas. C, C++, Fortran, Java, and Matlab are spoken here.

[...] Any effort to improving software has got to be taught to ALL engineers - not just the few S/W Engineers. It is the Aerospace PhD who gets to chose hardware and software policies, tools, and even procedures and architectures. S/W engineers do not get hired to do these things - they get hired to write code. That is the real world.

A Real Story...

From: Robert I. Eachus <rieachus@attbi.com>
Organization: Eachus Associates
Newsgroups: comp.lang.ada
Subject: Re: Ada -> C or C++ translator
Date: Wed, 03 Jul 2002

[In a thread about Ada to C or C++ translators. -- dc]

There was a project many years ago that had us all laughing when we found out what was going on. You could probably use the same technique...

The project manager and all of his team wanted to use Ada, but alas upper management would not let them. The code had to be in C so it would be portable. What did they do? Well they defined a number of C macros, then wrote all their code in (a subset of) Ada. Add an emacs macro which added the #defines and changed withs into includes, and they were all set. The project code was all written in Ada, but the check-in process compiled the C code as well, and the CM suite also checked overnight that the C and Ada versions produced identical output on the regression suite.

Sounds like a lot of work, but it took less than a week to get the first version of the toolset up and running. In fact it took some of the programmers longer to understand that if the Ada compile failed, there really was a bug in their code. The biggest complaints of course were about things that were valid in K&R C and some current C compilers "let you get away with." For example, returning -1 from a function defined to return a pointer...

In general the programming team felt that the "up front" cost through unit test balanced out. Getting initial clean compiles from both Ada and C took longer, but that was more than cancelled out by the strong typing benefits when interfaces needed to be modified. Of course, integration and test was scheduled for 8 months, and took three weeks. ☺

I know it sounds hard, but once they got started, the #defines were easy: [...]. And of course, there were areas of Ada that they had no intention of using such as tasking and exceptions, so those were not supported by the translation too.

Conference Calendar

This is a list of European and large, worldwide events that may be of interest to the Ada community. More information on items marked ♦ is available in the Forthcoming Events section of the Journal. Items marked with ☺ denote events with especially close relation to Ada.

The information in this section is extracted from the on-line *Conference announcements for the international Ada community* at: http://www.cs.kuleuven.ac.be/~dirk/ada-belgium/events/list.html on the Ada-Belgium Web site. These pages contain full announcements, calls for papers, calls for participation, programmes, URLs, etc. and are updated regularly.

2002

October 01-04	2002 IASTED **International Conference on Networks, Parallel and Distributed Processing, and Applications** (NPDPA'2002) Tsukuba Science City, Japan. *Topics include*: Distributed Processing, Distributed Real-time Systems, Parallel Processing, Parallel Programming, Parallel Computing Systems, Heterogeneous Computing, Compilers, Real Time and Embedded Systems, Applications, Fault Tolerance, Reusability, Reliability, etc.
October 02	8th IEEE **Workshop on Empirical Studies of Software Maintenance** (WESS'2002) Montreal, Quebec, Canada
October 03-04	2002 **International Symposium on Empirical Software Engineering** (ISESE'2002) Nara, Japan. *Topics include*: Evaluation of the readability of coding styles; Reports on the benefits derived from using software development environments; Development of predictive models of defect rates and reliability from real data; Industrial experience in process improvement; Quality measurement; Experience management; etc.
October 03-06	IEEE **International Conference on Software Maintenance** (ICSM'2002) Montreal, Canada. *Theme*: "Maintaining distributed heterogeneous systems". *Topics include*: Design for maintenance; Formal methods; Software reusability; Empirical studies; Programming languages; Maintenance and/or productivity metrics; Preventive maintenance; Tools and environments; Freeware and open source applications; Internet and distributed systems; Source code analysis and manipulation; Impact of new software practices; etc.
October 03-08	**Principles, Logics, and Implementations of high-level programming languages** (PLI'2002) Pittsburgh, USA
October 05-10	10th **International Conference on Architectural Support for Programming Languages and Operating Systems** (ASPLOS-X) San Jose, California, USA. *Topics include*: Interaction of operating systems, compilers, programming languages, and architectures; Case studies of hardware/software design in novel experimental systems; etc.
October 06-08	1st ACM SIGPLAN/SIGSOFT **International Conference on Generative Programming and Component Engineering** (GPCE'2002) Pittsburgh, PA, USA
☺ October 07-09	2nd **Workshop on Embedded Software** (EMSOFT'02) Grenoble, France. *Topics include*: System design and integration methodologies, Programming languages and software engineering, etc.
☺ October 08-11	**International Conference on Compilers, Architectures and Synthesis for Embedded Systems** (CASES'2002) Grenoble, France. Co-located with EMSOFT'2002. *Topics include*: Compilers and Operating Systems (new optimising compilers for embedded-domain constraints, compiler-controlled memory hierarchy management and smart caches, etc.); Architecture (synergy between extant parallel computing technologies, such as notations for expressing concurrency, and instruction level parallel processing, etc.); Tools and Methodologies (automated design and synthesis of application- or domain-specific processors, etc.); Applications; etc.
☺ October 13-16	21st **Symposium on Reliable Distributed Systems** (SRDS'2002) Osaka University, Suita, Japan. *Topics include*: Distributed systems with reliability, availability, security, safety, and/or real-time requirements; Distributed databases and transaction processing; Distributed objects and

middleware systems; Security and high confidence systems; Analytical or experimental evaluations of reliable distributed systems; etc. Includes among others the following event:

October 13   SRDS2002 - **International Workshop on Self-Repairing and Self-Configurable Distributed Systems** (RCDS'2002)

October 16-18   16th **Brazilian Symposium on Software Engineering** (SBES'2002) Gramado, Brazil. *Topics include*: Industrial applications of Software Engineering; Component-based Software Engineering; Theoretical Foundations of Software Engineering: Formal specification, refinement, software validation and verification; Methods, Techniques, Languages and Tools for Software Engineering; Software Maintenance; Software Quality; Software Reuse; Software verification, validation and testing; etc.

October 18-20   **Conference on Quality Engineering in Software Technology** (CONQUEST'2002) Nuremberg, Germany

October 22-25   4th **International Conference on Formal Engineering Methods** (ICFEM'2002) Shanghai, China

☺ October 27-31   21st **Digital Avionics Systems Conference** (DASC'2002) Irvine, California, USA. *Topics include*: avionics (flight critical systems, system engineering, open systems, software engineering, etc.), Air Traffic Management, etc.

October 28-30   12th International Conference on Software Quality (ICSQ'2002) Ottawa, Ontario, Canada

☺ October 28-11   4th **International Symposium on Distributed Objects and Applications** (DOA'2002) Irvine, California, USA. *Topics include*: Design patterns for distributed object design; Interoperability-supporting environments; Security, including authorisation and authentication; Reliable and fault-tolerant middleware; Real-time/Reflective middleware; Web Services and distributed objects, including SOAP interoperability and service discovery; Reports on Best Practice; etc.

October 28-11/01   9th IEEE **Working Conference on Reverse Engineering** (WCRE'2002) Richmond, Virginia, USA. *Topics include*: Software maintenance and evolution; Program comprehension; Software architecture extraction; Software migrations; Transitioning to product lines; Experience reports; Pre-processing and parsing; Software components; Reverse engineering tool support; UML and round trip engineering; Software metrics; etc.

♦ October 29   **Embedded Systems Club Autumn Conference** Swindon, UK. Combined Ada UK User Group. *Topics include*: any topic relevant to the embedded systems and/or Ada communities.

{US} November 03-05   **International Symposium on Computing in Object-Oriented Parallel Environments** (ISCOPE'2002) Seattle, Washington. Co-located with OOPSLA. *Topics include*: object-orientation in parallelism, distribution, high-performance applications and systems, components, linkage of XML, etc.

☺ November 04-08   17th Annual ACM SIGPLAN **Conference on Object-Oriented Programming**, **Systems, Languages, and Applications** (OOPSLA'2002) Seattle, WA, USA

November 05-08   1st **International Symposium on Formal Methods for Objects and Components** (FMCO'2002) Leiden, The Netherlands. *Topics include*: Specification, Verification, Validation, Programming languages, Design methodology, Tools, etc.

November 11-14   **International Conference on Formal Techniques for Networked and Distributed Systems** (FORTE'2002) Houston, Texas, USA. *Theme*: "Formal Methods for Protocol Engineering and Distributed Systems". *Topics include*: Formal approaches to concurrent/distributed Object-Oriented systems; Real-time and probability aspects; Verification and validation; Relations between informal and formal specification; Software tools and support environments; Practical experience and case studies; Corporate strategic and financial consequences of using formal methods; etc.

November 12-15   13th **International Symposium on Software Reliability Engineering** (ISSRE'2002) Annapolis, Maryland, USA. *Topics include*: Software testing and verification; Secure software engineering; Security testing and certification; Reliability of distributed systems; Standards and regulation; etc.

♦ November 15   DDC-I Course - **Introduction to Ada 95**. Phoenix, AZ, USA

November 18-21	1st **International Conference on Software Process Improvement** (ICSPI'2002) Adelphi, Maryland, USA. Deadline for early registration: October 18, 2002
November 18-22	ACM SIGSOFT 2002 10th **International Symposium on the Foundations of Software Engineering** (FSE-10) Charleston, South Carolina, USA. *Topics include*: Component-Based Software Engineering; Empirical Studies of Software Tools and Methods; Feature Interaction and Crosscutting Concerns; Generic Programming and Software Reuse; Software Engineering Tools and Environments; Software Reliability Engineering; Software Safety; Specification and Verification; etc.
November 20-22	14th **Nordic Workshop on Programming Theory** (NWPT'02) Tallinn, Estonia. *Topics include*: semantics of programming languages, programming language design and programming methodology, formal specification of programs, program verification, program construction, real-time and hybrid systems, tosols for program verification and construction, etc. Deadline for registration: October 21, 2002 (late)
November 25-26	**International Conference on Software Testing** (ICSTEST) Bilbao, Spain. *Theme*: "Testing Embedded Systems"
☺ December 02-04	8th IEEE **International Conference on Engineering of Complex Computer Systems** (ICECCS'2002) Greenbelt, Maryland, USA. *Topics include*: technologies for developing complex systems; means of avoiding, controlling, or coping with complexity; embedded real-time complex systems; distributed and network based complex software systems; design and analysis of complex software systems; formal methods for complex systems; techniques for component-based software development; etc.
December 03-05	15th **International Conference on Software & Systems Engineering and their Applications** (ICSSEA'2002) Paris, France
December 04-06	27th **Annual Software Engineering Workshop** (SEW27) Greenbelt, MD, USA. Co-located with ICECCS'2002. *Topics include*: Software quality assurance; Software engineering processes and process improvement; Real-time Software Engineering; Software maintenance, reuse, and legacy systems; etc.
December 04-06	9th **Asia-Pacific Software Engineering Conference** (APSEC'2002) Grand Mercure Broadbeach, Gold Coast, Queensland, Australia
December 08	2nd **Workshop on Industrial Experiences with Systems Software** (WIESS'2002) Boston, Massachusetts, USA. *Topics include*: Distributed Systems, Programming Environments and Tools, Fault Tolerance and High Availability, Real Time and Quality of Service, Middleware, Embedded Systems, etc.
♦ December 08-12	2002 ACM **SIGAda Annual International Conference** (SIGAda'2002) Houston, Texas, USA. Sponsored by ACM SIGAda, in cooperation with ACM SIGCSE, ACM SIGPLan, ACM SIGSoft and Ada-Europe (approvals pending). *Theme*: "Building Reliable Software". *Topics include*: Reliability needs and styles; Safety and high-integrity issues; Use of the Ada Distributed Systems Annex; Process and quality metrics; Testing and validation; Standards; Use of ASIS for new Ada tool development; Relationships between Ada and real-time Java; Mixed-language development; Ada in XML environments; Ada education; Use of Real-Time CORBA; Real-time networking/quality of service guarantees; Fault tolerance and recovery; Distributed system load balancing; Static and dynamic code analysis; Performance analysis; Debugging complex systems; Integrating COTS software components; System Architecture & Design.
December 09-11	4th **International Conference on Product Focused Software Process Improvement** (PROFES'2002) Rovaniemi (Arctic Circle), Finland. *Topics include*: Software Quality; Methods and Tools; Industrial Experiences and Case Studies; Best practices; Lessons Learned; Embedded Systems; etc.
December 09-11	5th **Symposium on Operating Systems Design and Implementation** (OSDI'2002) Boston, Massachusetts, USA. *Topics include*: distributed systems, parallel systems, embedded systems, the influence of hardware development on systems and vice-versa, etc.

December 10	Birthday of Lady Ada Lovelace, born in 1815. *Happy Programmers' Day!*
December 16-18	2002 **Pacific Rim International Symposium on Dependable Computing** (PRDC'2002) Tsukuba, Japan. *Topics include*: Design for system dependability; Fault-tolerant systems and software; Fault tolerance for parallel and distributed systems; Software and hardware reliability, verification and testing; Tools for design and evaluation of dependable systems; Application-specific dependable system (e.g., embedded systems, WWW servers, transaction processing); etc.
December 19-20	BCS-FACS **Formal Aspects of Security** (FASec) London, UK. Deadline for submissions: October 21, 2002
{IN} December 28-31	4th **International Workshop on Distributed Computing** (IWDC'2002) Calcutta, India. *Topics include*: Real Time and Embedded Systems; Distributed Objects and Algorithms; Language Translators for Distributed Environment; Fault-Tolerance and Reliability; etc.

2003

January 06-09	*Software Technology Track* of the 36th **Hawaii International Conference on System Sciences** (HICSS-36) Big Island of Hawaii, USA. *Includes mini-tracks on*: Experimental Software Engineering; Domain-Specific Languages; Distributed Object and Component-based Software Systems (Design Patterns for Distributed Systems, Middleware, Programming Languages and Environments for Distributed Object and Component Systems, etc.); etc.
January 08-11	**Workshop on Techniques for Trusted Components**. Prato (near Florence), Italy. Deadline for position paper submissions: October 1, 2002
{MX} ☺ January 13-15	8th IEEE Computer Society's **International Workshop on Object-oriented Real-Time Dependable Systems** (WORDS'2003) Guadalajara, Mexico
January 15-17	30th Annual ACM SIGPLAN-SIGACT **Symposium on Principles of Programming Languages** (POPL'2003) New Orleans, Louisiana, USA. *Topics include*: design, definition, analysis, and implementation of programming languages, programming systems, and programming abstractions.
{US} January 18	10th **International Workshop on Foundations of Object-Oriented Languages** (FOOL-10) New Orleans, Louisiana, USA. Following POPL'03. *Topics include*: the general area of foundations of object-oriented languages; language semantics, type systems, program analysis and verification, concurrent and distributed languages, etc.
February 03-05	9th **International Conference on Languages and Models with Objects** (LMO'2003) Vannes, France. *Topics include* (in French): *Programmation par objet et programmation par composant; Programmation par objet et modélisation par objets*; etc.
February 04-07	**Australasian Computer Science Conference** (ACSC'2002) Adelaide, South Australia. *Topics include*: Compilers, Concurrency, Distributed Systems, Embedded Systems, Fault Tolerance, Formal Methods, Object-Oriented Systems, Programming Languages, Real-time Systems, Reliability, Software Engineering, Trusted Systems, etc.
☺ February 05-07	11th **Euromicro Conference on Parallel Distributed and Network based Processing** (PDP'2003) Genoa, Italy. *Topics include*: Distributed Systems; Parallel Computer Systems; Models and Tools for Parallel Programming Environments; Advanced Applications (numerical applications with multi-level parallelism, real-time distributed applications, distributed business applications, etc.); Languages, Compilers and Runtime Support Systems (task and data parallel languages, object-oriented languages, scheduling and load balancing, task and object migration, etc.), etc. Special sessions on: Advanced Tools for Parallel and Distributed Programming; Parallel Real-time Systems; Memory Hierarchies; etc.
February 10-12	2nd **International Conference on Commercial Off-The-Shelf (COTS)-Based Software Systems** (ICCBSS'2003) Ottawa, Canada. *Theme*: "Multiple Paths, Multiple Solutions". Deadline for early registration: December 15, 2002
February 17	20th birthday of the first Ada standard (ANSI/MIL-STD-1815A), dated February 17, 1983

February 24-27	15th **Annual Software Engineering Process Group Conference** (SEPG'2003) Boston, Massachusetts, USA. *Theme*: "Assuring Stability in a Global Enterprise"
March 04-05	3rd **Workshop on Aspect-Oriented Software Development** (AOSD'2003) Essen, Germany. Deadline for submissions: November 15, 2002
March 05-06	2nd **International Workshop on Unanticipated Software Evolution** (USE'2003) Warsaw, Poland. *Topics include*: Formal methods, language concepts and implementation techniques for USE; USE support in programming languages, component models and related infrastructures; etc. Deadline for submissions: October 13, 2002
March 09-12	2003 ACM **Symposium on Applied Computing** (SAC'03) Melbourne, Florida, USA. *Includes tracks on Embedded Systems*: Applications, Solutions, and Techniques; Software Engineering: Applications, Practices and Tools; Programming Languages and Object Technologies; Parallel and Distributed Systems and Networking; etc.
{US} March 17-21	2nd **International Conference on Aspect-Oriented Software Development** (AOSD'2003) Boston, USA. *Topics include*: Applications; Software development methods; Reverse engineering and refactoring; Tools; Programming languages and implementation; Distributed systems; Composition, integration and evolution; etc. Deadline for submissions: October 4, 2002 (technical paper abstracts); October 11, 2002 (technical papers, practitioner reports); October 18, 2002 (workshops, tutorials); November 22, 2002 (demonstrations)
March 20-22	16th **Conference on Software Engineering Education and Training** (CSEET'2003) Madrid, Spain. *Theme*: "Software Engineering in Industry and University for the 21st Century". Deadline for submissions: October 1, 2002
March 26-28	7th **European Conference on Software Maintenance and Reengineering** (CSMR'2003) Benevento, Italy. *Topics include*: experience reports, enabling technologies, etc. Deadline for submissions: October 10, 2002
April 05-13	**European Joint Conferences on Theory and Practice of Software** (ETAPS'2003) Warsaw, Poland. *Event includes*: conferences from 7 to 11 April 2003, affiliated workshops on April 5-6 and 12-13, 2003. Includes among others the following events:
	April 07-11 ETAPS2003 - 12th **International Conference on Compiler Construction** (CC'2003). *Topics include*: compilation and interpretation techniques; run-time issues; language constructs and their implementation; modularization constructs and techniques for separate compilation; tools for compiler construction or language support, including debuggers, profilers, refactoring tools, etc. Deadline for submissions: October 18, 2002
	April 07-11 ETAPS2003 - 12th **European Symposium on Programming** (ESOP'2003). *Topics include*: design of programming languages and calculi; techniques, methods and tools for their implementation; exploitation of programming styles within different programming paradigms; automatic and manual methods for reasoning about programs; etc. Deadline for submissions: October 18, 2002
{US} April 07-11	10th IEEE **Symposium and Workshops on the Engineering of Computer Based Systems** (ECBS'2003) Huntsville, Alabama, USA. *Topics include*: Component-Based Design and Reuse; Middleware for Embedded Systems; Applied Formal Methods; Education and Training; Embedded Systems; Evolution, Reengineering and Legacy Systems; Reliability, Dependability, Safety and Security; Verification and Validation Standards; etc. Deadline for submissions: October 28, 2002
☺ April 22-26	**International Parallel and Distributed Processing Symposium** (IPDPS'2003) Nice, France. *Topics include*: Parallel and distributed software, including parallel programming languages and compilers, operating systems, runtime, middleware, libraries, programming environments and tools for parallel and distributed computing, etc. Deadline for submissions: October 4, 2002 (manuscripts); October 20, 2002 (tutorial, industrial track submissions). Includes among others the following event:

April 22-26   IPDPS2003 - 8th **International Workshop on Formal Methods for Parallel Programming: Theory and Applications** (FMPPTA'2003) Deadline for submissions: October 28, 2002

☺ May 03-10   25th **International Conference on Software Engineering** (ICSE'2003) Portland, Oregon, USA. Deadline for submissions: October 4, 2002 (Software Engineering Education and Training Track, experience reports, tutorials, panels, workshops); January 20, 2003 (demos, posters, doctoral symposium)

{JP} ☺ May 14-16   6th IEEE **International Symposium on Object-oriented Real-time distributed Computing** (ISORC'2003) Hakodate, Hokkaido, Japan

{US} May 19-22   23rd **International Conference on Distributed Computing Systems** (ICDCS'2003) Providence, Rhode Island, USA. *Topics include*: Middleware; Fault Tolerant and Dependable Systems; Real-time and Embedded Systems; Software Engineering and Formal Methods; etc.

{US} June 09-14   28th **Annual USENIX Technical Conference** (USENIX'2003) San Antonio, Texas. *Topics include*: applications, architecture, implementation, and performance of modern computing systems; Reliability and QoS; Interoperability of heterogeneous systems; special FREENIX track on freely redistributed technology; etc. Deadline for submissions: November 18, 2002

♦ June 16-20   8th **International Conference on Reliable Software Technologies - Ada-Europe'2003**. Toulouse, France. Sponsored by Ada-Europe, in cooperation with ACM SIGAda (approval pending). *Topics include*: Management of Software Development and Maintenance; Software Quality; Software Development Methods and Techniques; Software Architectures; Tools; Kinds of Systems; Applications; Ada Language and Tools; Ada Experience Reports; Education and Training; Case Studies and Experiments; and a special session on Avionics and Space, including the use of Ada in this realm. Deadline for submissions: October 31, 2002 (papers, extended abstracts, tutorials, workshops)

{BR} June 16-20   ACM/IFIP **International Middleware Conference** (Middleware'2003) Rio de Janeiro, Brazil. *Theme*: "Information Systems for a Connected Society". *Topics include*: Distributed real-time and embedded (DRE) middleware platforms; Reliable and fault-tolerant middleware platforms; Formal Methods applied to middleware; Novel paradigms, APIs, and languages for distributed systems; etc. Deadline for submissions: October 20, 2002 (workshop proposals); December 22, 2002 (papers); February 1, 2003 (tutorial proposals); February 20, 2003 (work in progress papers, workshop papers, posters)

♦ September 15-19   12th **International Real-Time Ada Workshop** (IRTAW-12) Pousada Monte de Sta. Luzia, Viana do Castelo, Portugal. (http://www.hurray.isep.ipp.pt/irtaw2003) Topics include: Ada language revision process; lessons learned from the industrial use of Ada 95; resilient, distributed, real-time systems; OOP paradigm and impact on real-time programming; real-time specification for Java and inter-operation with Ada

December 10   Birthday of Lady Ada Lovelace, born in 1815. *Happy Programmers' Day!*

2004

{ES} March 29- April 02   **European Joint Conferences on Theory and Practice of Software** (ETAPS'2003) Barcelona, Spain. *Event includes*: conferences from March 29 to April 2, 2004, affiliated workshops on 27-28 March and 3-4 April 2004 Deadline for submissions: December 1, 2002 (workshop proposals)

Preliminary Call for Participation – SIGAda 2002

8-12 December 2002, Houston, Texas, USA
Sponsored by ACM SIGAda
http://www.acm.org/sigada/conf/sigada2002

(Approval pending by ACM)

Constructing reliable software is an engineering challenge. The application of methods, tools, and languages interrelate to make the challenge easier or more difficult. This conference focuses on the interaction between these three aspects of software engineering, especially how features in a language such as Ada drive the tools, methods, and ultimately correctness, reliability, and quality of the resulting software. Especially welcome are papers that analyze Ada with respect to these factors or in comparison with other languages. This conference will gather industrial experts, educators, software engineers, and researchers interested in developing and testing reliable software. Technical or theoretical papers as well as experience reports with a focus on Ada are solicited. Possible topics include but are not limited to:

- Reliability needs and styles
- Safety and high integrity issues
- Use of the Ada Distributed Systems Annex
- Process and quality metrics
- Testing and validation
- Standards
- Use of ASIS for new Ada tool development
- Relationships between Ada and real-time Java
- Mixed-language development
- Ada in XML environments

- Ada education
- Use of Real-Time CORBA
- Real-time networking/quality of service guarantees
- Fault tolerance and recovery
- Distributed system load balancing
- Static and dynamic code analysis
- Performance analysis
- Debugging complex systems
- Integrating COTS software components
- System Architecture & Design

How You Can Contribute

SIGAda 2002 is interested in receiving contributions in six major categories**.** Contributions from students are actively solicited. **Technical Articles** present significant results in research, practice, or education. These papers will be double-blind refereed and published in the Conference Proceedings. Papers should not exceed 5000 words (equivalent to approximately 10 pages, typeset 10-point on 16-point spacing). **Extended Abstracts** discuss current work for which early submission of a full paper may be premature. If your abstract is accepted, you will be expected to produce a full paper, which will appear in the proceedings. Extended abstracts will be competitively reviewed. Clearly state the contribution of the work being described, its relationship with previous work by you and others (with bibliographic references), results to date, and future directions. Please do not exceed 2500 words (equivalent to approximately 5 pages typeset 10-point on 16-point spacing). **Experience Reports** present timely results on the application of Ada and related technologies to the design and implementation of applications such as the following: avionics, aerospace, automobile, command and control, consumer electronics, process control, transportation, trading systems, energy, medical systems, simulation, telecommunications, etc. Such reports will be selected on the basis of the interest of the experience presented to the community of Ada practitioners. You are invited to submit a 1-2 page description of the project and the key points of interest of project experiences. Descriptions will be published in the final program or proceedings, but a paper will not be required. **Workshops** are focused work sessions, which provide a forum for knowledgeable professionals to explore issues, exchange views, and perhaps produce a report on a particular subject. A list of planned workshops and requirements for participation will be published in the SIGAda 2002 Advance Program. Workshop proposals will be evaluated by the Program Committee and selected based on their applicability to the conference and potential for attracting participants. Proposals should state the problem or issue to be addressed, the coordinator(s), and criteria for participant selection. **Panel Sessions** gather a group of experts on a particular topic who present their views and then exchange views with each other and the audience. Panel proposals should be 1-2 pages in length, identifying the topic, coordinator, and potential panelists. **Tutorials** offer the flexibility to address a broad spectrum of topics relevant to Ada, and those enabling technologies which make the engineering of Ada applications more effective. Submissions will be evaluated based on relevance, suitability for presentation in tutorial format, presenter's expertise, and past performance. Tutorial proposals should include the expected level of experience of participants, an abstract or outline, the qualifications of the instructor(s), and the length of the tutorial.

Please submit **Technical Articles, Extended Abstracts, Experience Reports, Workshop proposals, and Panel Sessions** to the Program Chair, John McCormick <McCormick@cs.uni.edu> and **Tutorial proposals** to the Tutorials Chair, David Cook <david.cook@hill.af.mil>. Please submit questions on the conference to the Conference Chair, Salih Yurttas <yurttas@cs.tamu.edu>.

Deadline for Tutorial submissions: 6 May 2002; Deadline for other submissions: 3 June 2002

See SIGAda 2002 Home Page for details: http://www.acm.org/sigada/conf/sigada2002

	12th International Real-Time Ada Workshop **IRTAW-12** **15-19 September 2003** **Pousada Monte de Sta. Luzia, Viana do Castelo** **Portugal**

First Announcement

Over the last 15 years, the series of the **International Workshop on Real-Time Ada Issues** have provided focus for identifying issues with Ada 83 and 95; for proposing solutions to those problems; and for evaluating proposed changes to the language standard.

Since the standardization of Ada 95, the **IRTAW** series have assisted in the review of the real-time related chapters of the *Guide for the Use of the Ada Programming Language for High Integrity Systems* (ISO/IEC TR 15942:2000) and have developed and promoted the Ravenscar Tasking Profile. With the advent of Java and the development of Real Time Specification for Java, the Workshop has begun to consider the integration of embedded Ada and Java systems, and their interoperability.

In keeping with this tradition, the goals of **IRTAW-12** will be to:

- Examine the shape and the status of the language amendments proposed or promoted by previous **IRTAW** editions, with respect to the ongoing Ada language revision process, as well as the demand for further enhancements

- Consider the lessons learned from industrial experience with using Ada 95 in general and the Ravenscar Profile in particular in actual real-time projects

- Examine and develop paradigms for using Ada 95 for real-time distributed systems, including issues of robustness as well as of hard, flexible and application-defined scheduling

- Consider the impact of the OOP paradigm on multi-threaded, possibly distributed, real-time systems

- Review the status and contents of the *Guide for the Use of the Ada Ravenscar Profile in High Integrity Systems* (to become an ISO/IEC TR) and examine the issues of certifying software systems developed using Ada 95

- Examine the status of and the progress with the Real-Time Specification for Java and consider user experience with the reference implementation and with issues of interoperability with Ada in embedded real-time systems.

Participation at **IRTAW-12** is by invitation following the submission of a position paper addressing one or more of the above topics. Position papers should not exceed six pages. All accepted papers will appear, in their final form, in the Workshop Proceedings, which will be published as a special issue of Ada Letters (ACM Press).

Submit position papers, in PDF, to the Program Chair by e-mail at: **tullio.vardanega@math.unipd.it**

Program Committee:

Ben Brosgol, Alan Burns, Brian Dobbing, Michael Gonzalez Harbour, Mike Kamrad, Stephen Michell, Luis Miguel Pinho (Local Chair), Juan Antonio de la Puente, George Romanski, Joyce Tokar, Tullio Vardanega (Program Chair), Andy Wellings.

Important Dates

Receipt of Position Paper: 1 June 2003   Final Copy of Paper: 1 November 2003
Notification of Acceptance: 30 June 2003   Workshop Date: 15-19 September 2003

http://www.hurray.isep.ipp.pt/irtaw2003

Customizing UML for the Development of Distributed Reactive Systems and Ada 95 Code Generation

Michael Kersten, Jörg Matthes, Christian Fouda Manga, Stephan Zipser, Hubert B. Keller

Forschungszentrum Karlsruhe, D-76021 Karlsruhe, Germany. Homepage: www.iai.fzk.de

Abstract

Distributed, reactive software systems, e.g. process control tools, can be modelled with the Unified Modelling Language (UML). Recently such UML models are used to generate source code automatically. Because of the complexity of UML, it is necessary to restrict the usage of its constructs by defining UML profiles to allow the automatic generation of source code. In this paper a UML profile for the design and implementation of distributed, reactive systems and an associated mapping to Ada 95 source code are introduced. Further, our experiences with the chosen approach are discussed.

Keywords: UML, automatic code generation, Ada.

1 Introduction

In our research group at the Forschungszentrum Karlsruhe we optimize industrial processes using methods like infrared thermography, fuzzy control and image processing. Since available process control tools have not been able to implement complex algorithms in a time effective and comfortable way we decided to overcome this situation with the development of the process control tool Inspect 2 [1].

First of all, the Inspect 2 system can be classified as a reliable, distributed, reactive system. Second, the area of research implies high maintainability requirements. To cope with this situation we decided to use a special development process, which combines the good maintainability characteristics of UML with the advantages of Ada 95 of reliability aspects.

In cooperation with the Company Aonix, Karlsruhe, we defined a UML profile for distributed reactive systems and developed ACD (Architecture Component Development) templates for code generation to Ada 95 source code with the UML tool Software through Pictures (StP) 8.0[1].

ACD allows the user to define code generation templates for any implementation language using a script language. In theory it is possible to define own stereotypes and tagged values in StP, and to access these stereotypes by the code generator, and therefore almost any mapping between UML and the implementation language of choice can be defined. The details concerning the usage and implementation of the templates for the generation of Ada 95 source code can be found in [2]. The described UML profile and the associated templates developed together with Aonix in the Inspect 2 project are part of the latest release StP 8.3.

This report describes the UML profile and reviews our experiences with code generation from UML to Ada 95 based on StP.

2 The process Control Tool Inspect 2

In order to use Inspect 2 as a running example we will first of all give a brief description of its architecture (see Figure 1).
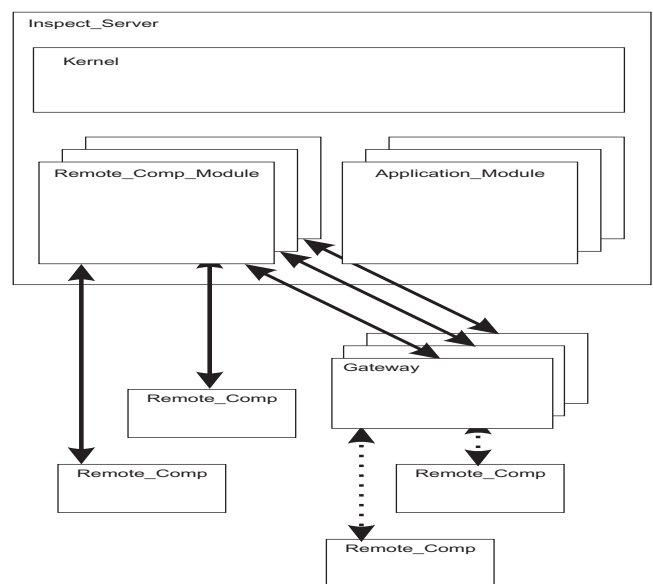

Figure 1: The structure of Inspect 2

Inspect 2 is a client server system with the Inspect server managing the collection, distribution and archiving of data.

---

[1] Parts of the UML model, describing the communications subsystem, have also been mapped to Java source code

Additionally, the server is able to schedule various analysis and control tasks. The clients, communicating with the server via TCP/IP, are responsible for the measurement of process values and for the communication with other systems, e.g. the process control system. Special clients (Gateways) allow the communication with non-TCP/IP based systems. Clients also act as distributed/remote human machine interfaces.

Each client communicates with the server via an own associated task (Remote_Comp_Module) within the server, which maintains the communication and caters for a reliable data transfer. The application logic of Inspect (i.e. data processing) is defined in Application_Module tasks. All modules can be customized by configuration and are scheduled by the Inspect server.

At present instances of Inspect 2 are used in the field of industrial combustion control and sensor networks.

3 The UML Profile

As we already motivated in [3], an advantage in the use of UML is its possibility of customization. Since UML is a complex family of languages (see [4]) with a large amount of concepts, it is generally convenient to restrict these to the minimum required by the application domain. Otherwise problems in the mapping of UML diagrams to Ada 95 code are induced. Due to the fact that there are no complete formal semantics for UML, we decided to define a UML profile and give denotational semantics to it by mapping its concepts to the concepts of Ada 95.

For better understanding we do not use the UML notation for profiles (see [5]), but we define the concepts of our profile informally and illustrate them using parts of our running example Inspect 2. The structure of the profile is given in Figure 2. It uses class diagrams and statechart diagrams because they are able to describe distributed reactive systems completely. Class diagrams model the structural and functional view of the system to be described. Statechart diagrams model the behavioural view.

In the following we will describe the concepts of our UML profile and illustrate them using our running example Inspect 2 (see Figure 3).

3.1 Packages

In UML, the concept of Packages is used in the Model Management Part for the grouping of model elements (refer to [5], pp. 3-17). Since in standard UML subsystems are behavioral units rather than structural units (refer to [5], pp. 3-21) we introduce the stereotype subsystem for packages. In this sense in our

profile a subsystem is a package that groups model elements, which belong to structural parts of the system.

In our profile only packages with the stereotype subsystem are allowed. Subsystems may contain subsystems and for simplicity the total system is modeled by a subsystem too.
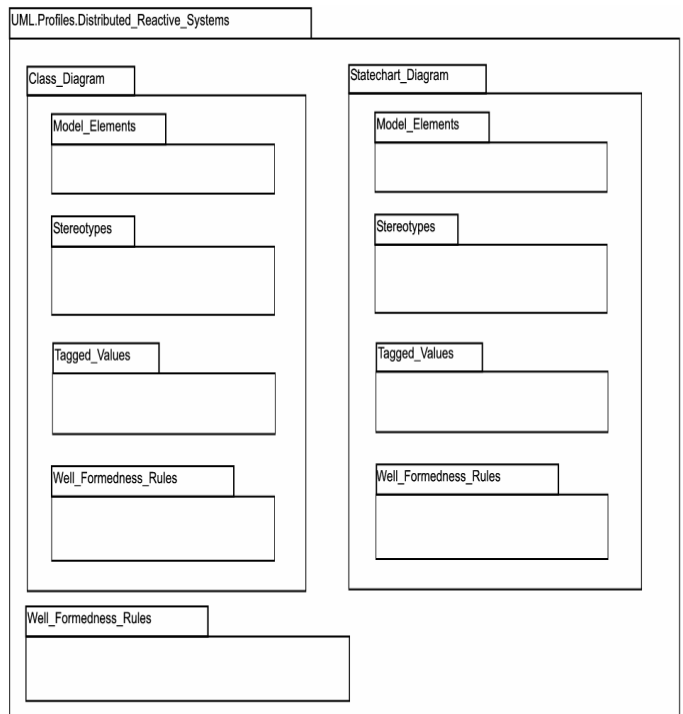

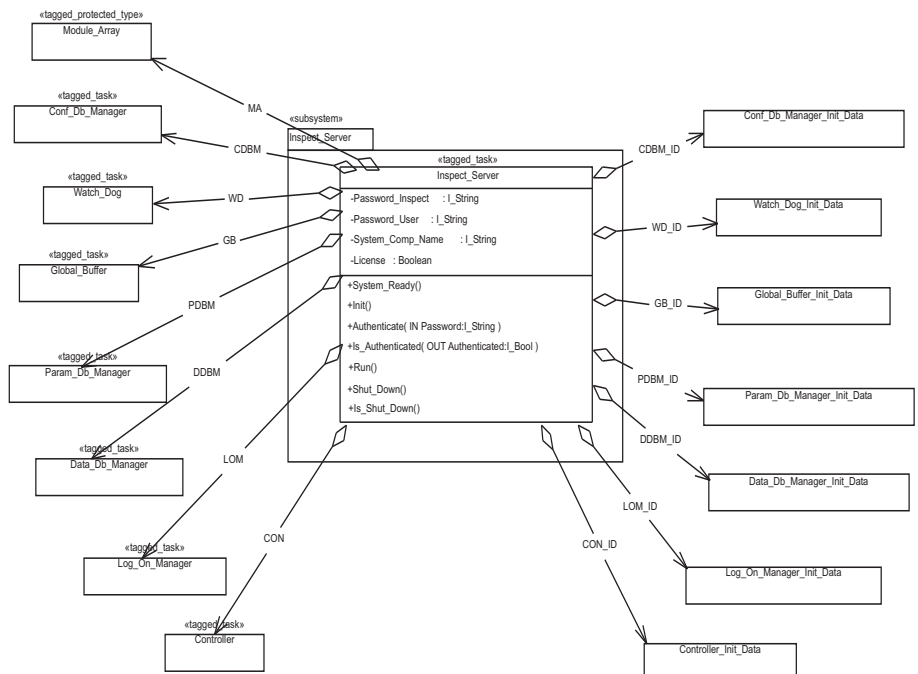**Figure 2: The structure of our profile**

Figure 3: The structure of Inspect 2 as class diagram

Figure 3 shows an example for a package with stereotype subsystem: the package Inspect_Server.

3.2 Classes

In UML, Classes (refer to [5], pp. 3-34) follow the concept of abstract data types, i.e. they are container for typed attributes and operations over these. In our profile attributes and operations can have public or private visibility (as described in [5],p. 3-40 and 3-43). The protected visibility property is not allowed in our profile.

In order to model the concurrent behavior of distributed reactive systems we introduced the stereotypes Tagged_Protected_Type and Tagged_Task for classes:

A class with the stereotype Tagged_Protected_Type is a special case of a passive class. It is a storage class with a mechanism for the synchronization of concurrent access of multiple tasks.

The stereotype Tagged_Task defines an active class with a light-weight concurrent flow of control. In contrast to the standard passive classes our tagged_-task class has an own flow of control. To describe this flow of control, each Tagged_Task is associated with a statechart. For the connection of the class and the statechart diagram, the

- Private methods of the class allow local abstraction and the expression of a behaviour, which is hidden to the environment (other classes).

- Methods of classes are non-blocking (i.e. execution is locally without any further entry call or accept statement). Otherwise they could cause deadlocks.

A further new concept for classes with the Tagged_Task or Tagged_Protected_Type stereotype is the tagged value Timed_Call_Supported for public operations. These so-called timed operations have different semantics than normal public operations. The call of a normal public operation of a Tagged_Protected_Type or a Tagged_Task leads to the situation that the caller is blocked until the operation is finished. If in the case of a Tagged_Task the associated statechart has no outgoing transition in the current state, which is annotated with the corresponding event, the caller is blocked until the Tagged_Task reaches a state with this predicate. In the worst case deadlocks and livelocks can occur, even though the called method itself is non-blocking. A simple and effective way of avoiding liveness problems is the usage of timed operations. In our profile they have
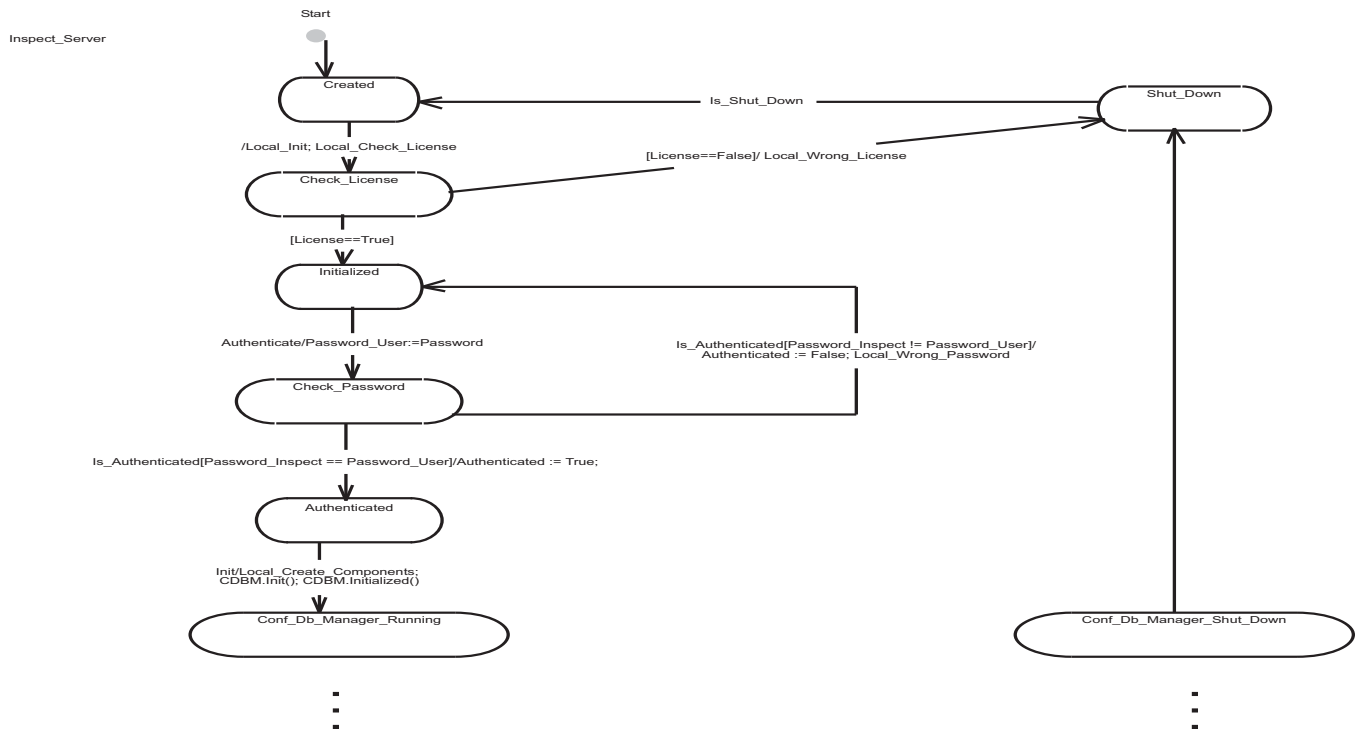

Figure 4: Fragment of the statechart of the class inspect_server

following well-formedness rules must hold:

- Each public method of the class must be used as an event in the associated statechart diagram and vice versa.

- Guard conditions in the statechart only refer to the attributes of the class diagram.

- The change of values of attributes of the class is modelled in the associated statechart diagram as actions.

- Public methods of the class are external events, which allow communication between different classes.

the semantic that the caller of the operation is only blocked until either the operation call is accepted or the timeout of the operation is expired. All timed operations have an extra in-parameter for specifying the maximum waiting time and an extra out-parameter informing the caller if the call was accepted (and processed) or not. These two parameters are implicitly defined in the class. On the caller side both parameter values need to be attached.

Figure 4 shows a fragment of the associated statechart diagram of the class Inspect_Server.

Between the states Initialized and Check_Password there are two interesting transitions. The transition from Initialized to Check_Password is triggered by the event Authenticate. After this event the attribute Password_User is set to Password.

Password is the value of the concrete parameter of the method Authenticate. The parameter is only defined in the class diagram (see Figure 4) and not in the statechart[2]. The transition from Check_Password to Initialized gives an example of a complex transition.

If the event Is_Authenticated occurs the guard condition [Password_Inspect!=Password_User] is evaluated and the transition takes place if the result is true (i.e. the password is wrong). In that case two actions are executed. The first action changes the classes attribute Authenticated to the value False. The second action is the execution of the local method Local_Wrong_Password. This is another interesting concept of our approach. The user may specify actions in the statechart, which are not member of the associated class. These are mapped to locally defined Ada procedures in the body and represent an additional mechanism of local abstraction. The semantic of the local procedure is defined in the implementation and does not matter in the design. The use of this concept increases the clarity and understandability of the statechart diagram. It is important to avoid the definition of blocking local procedures in the code since this leads to deadlocks. Since local procedures are used as a concept of abstraction between the design and implementation level, we deliberately leave the avoidance of blocking in the responsibility of the developer. Thus our approach does not provide a method to ensure the non-blocking definition of local procedures. We would therefore recommend not using infinite loops and external operation calls. The latter cannot always be considered, because the main purpose of local procedures is the use of operating system functions.

3.3 Associations

UML consists of a rich set of association types. In our profile only the following types of associations are necessary:

- Generalization
- Aggregation
- Composition

For the use of the inheritance features, the generalization association (see [5], p. 3-79) is included in the profile. Additionally our profile includes the association type composition as described in [5] on page 3-74.

The most important association type in our profile is the aggregation association (see [5], p. 3-79). In contrast to standard UML the only restriction is the deprecation of association classes for the aggregation.

---

[2] By convention the parameters of classes event can be omitted if the name of the event is unique in the class

For the aggregation our profile contains the stereotype Owner. The so-called ownership aggregation models a close binding between the associated classes. One side of the association has the role of the owner and is responsible for the creation and destruction of the class on the other side.

3.4 Well-formedness Rules

In order to avoid liveness problems we decided to use only hierarchic visibility between classes.

4   Mapping the Concepts to Ada 95

4.1 Packages

UML Packages with the stereotype Subsystem are mapped to folders in the project-folder.

4.2 Classes

The classes of the profile are mapped to Ada 95 packages with the following content:

- A record with one component for each attribute of the class and one component for each aggregation of the class.
- A public procedure for each public operation of the class.
- A private procedure for each private operation of the class.

4.2 Tagged_Protected_Type

The classes with stereotype Tagged_Protected_Type are mapped to an Ada 95 package containing the following elements:

- A record with one component for each attribute of the class and one component for each aggregation of the class.
- A private protected type Class_Name_Pt with one entry for each class operation (with isomorphic signatures).
- A public procedure Operation_Name for each class operation with a signature isomorphic to the class operation (In the procedures body the corresponding entry of the protected type Class_Name_Pt is called.).
- A public procedure Timed_Operation_Name for each class operation with tagged value Timed_Call_Supported, which has an isomorphic signature to the class concatenated with the additional parameters for timing.

These are (in:Time:duration,out:timed_out:boolean) (In the body of the procedure the corresponding entry of the protected type Class_Name_Pt is called as a timed operation.).

4.3 Tagged_Task

The classes with stereotype Tagged_Task are mapped to an Ada 95 package which includes the following elements:

- A record with one component for each attribute of the class and one component for each aggregation of the class.

- A variable Current_State of an enumeration type Class_Name_Event_Type containing all states of the associated statechart diagram.

- A private task type Class_Name_Task with one entry for each event in the statechart diagram. The entries have a signature isomorphic to the corresponding class operation.

- A public procedure Operation_Name for each class operation with a signature isomorphic to the class operation (In the procedures body the corresponding entry of the task type Class_Name_Task is called.).

- A public procedure Timed_Operation_Name for each class operation with the tagged value Timed_Call_Supported which has a signature isomorphic to the class concatenated with the additional parameters for timing. These are (**in**:Time:duration,**out**:timed_out:boolean) (In the body of the procedure the corresponding entry of the task type Class_Name_Task is called as a timed operation.).

- A statechart implementation using select-statements depending on the value of the variable Current_State.

- A private procedure for each action in the statechart diagram with a signature isomorphic to the action.

4.4 Code examples

In the section above we described the mapping of the UML concepts to Ada 95. For a better impression to the reader we will give a detailed code example. First, we will show the Ada specification file of the Inspect_Server_Pkg, whose UML representation was discussed in section 3[3].

In the specification the package Inspect_Server_Pkg is introduced:

```
package Inspect_Server_Pkg is
    type Inspect_Server is new
        ACD_Runtime.Active_Base_Class.ActiveLimitedInstance
        with private;
    type Inspect_Server_Cptr is access all
        Inspect_Server'class;
    type Inspect_Server_Ptr is access all Inspect_Server;
    ...
```

It consists of the derived type Inspect_Server with a pointer Inspect_Server_Ptr and a class wide pointer Inspect_Server_Cptr. The type ACD_Runtime.Active_Base_Class.ActiveLimitedInstance is an empty tagged record type which represents the type active class in the profile and was adopted from the standard ACD code generation templates. We found it usefull, because it enables the user

---

[3] For better readability we reformatted the automatically generated file and removed irrelevant comments.

to define global properties of active classes, if necessary. In the private part the type is redefined.

Next, the type Inspect_Server_Event_Type defines the set of events of the class Inspect_Server:

```
type Inspect_Server_Event_Type is
    (Is_Authenticated_Event, Is_Shut_Down_Event,
    System_Ready_Event, Shut_Down_Event, Init_Event,
    Authenticate_Event, Run_Event);
```

The correspondence between events, class operations and entries of the task are shown in the body. The next parts in the specification are the constructor and destructor operations:

```
--Constructor Operations-----------------------------------------
procedure Initialize(Acc_This : Inspect_Server_Cptr);
function Create return Inspect_Server_Ptr;
--Destructor Operations-----------------------------------------
    procedure Finalize(
        Acc_This : in out Inspect_Server);
    procedure Free (
        Acc_This : in out Inspect_Server_Cptr);
```

Next, the class operations are specified. Since the mapping principle is the same for each operation, we show only two examples:

```
--Operations-----------------------------------------------------
    procedure System_Ready(
        Acc_This : access Inspect_Server);
    -- Timed_Call_Supported:
    procedure Timed_System_Ready(
        Acc_This : access Inspect_Server;
        Timeout : duration;
        Timed_Out : out Boolean);
    procedure Init(Acc_This : access Inspect_Server);
    -- Timed_Call_Supported:
    procedure Timed_Init(
        Acc_This : access Inspect_Server;
        Timeout : duration;
        Timed_Out : out Boolean);
```

The reader may have recognized that two versions of each operation are defined. This reflects that for these operations the tagged value Timed_Call_Supported is set in the UML model. The timed versions of the procedures have prefix Timed_ attached. In the private part of the specification, the static part of the associated state-machine of the active class Inspect_Server is defined:

```
private
package Inspect_Server_State_Machine is
--State Type----------------------------------------------------
    type State_Type is (
        ST_Authenticated,
        ST_Check_Password, ...);
end Inspect_Server_State_Machine;
```

This is the type Inspect_Server_State_Machine, which models the set of states of the state-machine. The transitions of the state-machine are defined in the body. Since each active class has an own thread of control a task type is necessary:

```
task type Inspect_Server_Task(Acc_This : access
    Inspect_Server'class) is
--State Machine Operations-------------------------------------
    entry Take_Is_Authenticated_Event(
        Authenticated : out Types.I_Bool);
```

```
   ...
     entry Take_Run_Event;
   end Inspect_Server_Task;
```

For each event of the state-machine the task type has a corresponding entry. To increase the readability of the code, the prefix Take_ is attached to the entries. The example above shows only two entries. As stated earlier, the type Inspect_Server is redefined in the private part:

```
type Inspect_Server is new
   ACD_Runtime.Active_Base_Class.ActiveLimitedInstance with
   record
      Tsk : Inspect_Server_task(
         Acc_This => Inspect_Server'access);
   --Attributes-----------------------------------------
      Password_Inspect : Types.I_String;
      ...
   --Relations------------------------------------------
      WD_Part : Watch_Dog_Pkg.Watch_Dog_Cptr;
      MA_Part : Module_Array_Pkg.Module_Array_Cptr;
      ...
   end record;
end Inspect_Server_Pkg;
```

The first component of the tagged record is of the type of the previously defined task type. This shows the encapsulation of the task in the tagged record, which is the central concept of the active class implementation. The second component shown in the example is the attribute Password_Inspect. The other attributes are not shown. After that, two examples of relations to other classes are shown. The prefixes WD and MA are the names of the relations in the class diagram (refer to Figure 3). By convention each relation attribute is named by the relation name followed by _Part.

After this short overview of the Ada specification of the Inspect_Server_Pkg we will have a look to the associated Ada body file:

```
package body Inspect_Server_pkg is
 package body Inspect_Server_State_Machine is
 end Inspect_Server_State_Machine;
```

The code fragment above shows the implementation of the state-machine type. Since it is an enumerated type the most information is still defined in the Ada specification.

More interesting is the implementation of the Inspect_Server_Task:

```
task body Inspect_Server_Task is
   This : Inspect_Server'class renames Acc_This.all;
--State Machine--------------------------------------
   currentState:Inspect_Server_State_Machine.State_Type :=
   Inspect_Server_State_Machine.ST_Start;
   use Inspect_Server_State_Machine;
```

For readability reasons the access to self (Acc_This.all) is renamed by the keyword This. After that, the variable currentState is declared and set to the initial state.

Next, the local procedures are defined. As shown in Figure 4, local procedures implement the actions of the UML statecharts which model the behavior of active classes. For example we show a fragment of the implementation of the

local procedure Local_Create_Components, which creates all components of the active class Inspect_Server:

```
procedure Local_Create_Components is
begin
   --Create Module_Array
   This.MA_Part := Module_Array_Pkg.Create.all'access;
   --Create Watch_Dog in the Pkg
   Watch_Dog_Pkg.The_Watch_Dog_Ptr :=
      Watch_Dog_Pkg.Create.all'access;
   This.WD_Part := Watch_Dog_Pkg.The_Watch_Dog_Ptr;
   ...
end Local_Create_Components;
```

The following code fragment shows the implementation of the state-machine. For a good understanding a comparison of the implementation with the UML statechart diagram (refer to Figure 4) is useful. The core of the implementation is a case statement over the variable currentState enclosed in a loop. The case statement branches (via when-statements) over each possible state of the state-machine. Since the variable currentState enumerates over all states of the state-machine it is an invariant of the implementation so in the focus of the loop statement exactly one when-expression evaluates to true. This implies that the order of the when-statements is inessential. We only show the states Initialized and Check_Password in our example because they include all relevant concepts of our state-machine implementation:

```
loop
   case currentState is
      ...
      when Inspect_Server_State_Machine.
         ST_Check_Password =>
         -- Activity:
         null; -- user defined code to be added here
         select when (This.Password_Inspect =
            This.Password_User ) or
            (This.Password_Inspect /= This.Password_User )=>
            accept Take_Is_Authenticated_Event(
               Authenticated : out Types.I_Bool) do
               if This.Password_Inspect =
                  This.Password_User then
                  Authenticated := True;
                  currentState := ST_Authenticated;
               elsif This.Password_Inspect /=
                  This.Password_User then
                  Authenticated := False;
                  Local_Wrong_Password;
                  currentState := ST_Initialized;
               end if;
            end Take_Is_Authenticated_Event;
         end select;
      when Inspect_Server_State_Machine.ST_Initialized =>
         -- Activity:
         null; -- user defined code to be added here
         select when ( TRUE ) =>
            accept Take_Authenticate_Event(
               Password : in Types.I_String) do
               if TRUE then
                  This.Password_User:=Password;
                  currentState := ST_Check_Password;
               end if;
            end Take_Authenticate_Event;
         end select;
      ...
   end case; -- end state case
end loop;
```

Assume the variable currentState has the value ST_Check_Password. Thus Inspect_Server_State_Machine.-ST_Check_Password evaluates to true and the other when-expression evaluates to false. Then the code after the first

---

when-statement is executed. Because there is no entry action defined in the state Check_Password, there is a null-statement after the comment *–Activity*. Next, the implementation of the outgoing transitions of the state Check_Password is shown. Both are triggered by the event Is_Authenticated (for both guard conditions Take_Is_Authenticated_Event is accepted). When this event occurs (the operation Is_Authenticated is called) the guards are evaluated in the if-elsif-combination. If the if-branch evaluates to true, authenticated ist set to true and the next currentState is set to ST_Authenticated. Otherwise authenticated is set to false and the next currentState is set to Initialized.

Assume the current state is Initialized, there is one possible outgoing transition. In contrast to the transitions above this one has no guard condition. In the select-statement the expression when (TRUE) is used which is constantly true.

Whenever the event Take_Authenticate_Event occurs (the operation Authenticate is called), the guard if TRUE evaluates to true and the action code is executed. This means, that the attribute This.Password_User is set and the next value of the variable currentState is set to ST_Check_Password.

5 Conclusions

The above-described UML profile for distributed reactive systems was developed at the Forschungszentrum Karlsruhe for internal use. We used it for the development of the process control tool Inspect 2 which we used as running example in this paper.

A measurement of the systems complexity[4] did not take place yet, but we can provide some estimates in order to get an idea of the projects effort. The UML model is decomposed into 19 subsystems with 31 class diagrams and 13 statechart diagrams. These are mapped to nearly 150 packages of Ada 95 source code.

Since the project was the first using the presented approach there was an overhead in the early project phases. This was caused by the need of defining the UML profile and the associated mapping to Ada 95. Another cost factor was the analysis of the existing design tools, their installation and test. Additionally the customization of the Ada 95 templates for code generation increased the start up costs.

After finishing the projects foundations we have to point out that most of the effort of the implementation phase was drawn in the earlier design phase. Even some effort of the documentation was done previously.

During the development of the system there was one very important philosophy in the developers mind: whenever a

---

[4] Complexity measurement of software is an interesting but elaborate scheme, in particular in the domain of the object-oriented analysis and development. We use the notion of complexity in its natural sense, since the use of a formal complexity measure is no benefit for the reader for the understanding of our presented approach.

change could be made in the design instead of the implementation it was made in the design. This strong focus on the design phase implied a good understanding of the system to be developed (which reduces logical faults) and a very tight relationship between the design and implementation model.

The common coding errors could be minimized. Therefore the implementation phase was very short. Additionally the testing phase[5] was decreased extremely. This fact was induced by the excellent debug capabilities of the approach. Whenever a logical failure occurred during the testing phase it was easy to find the error because its area could be determined using the design model and specifically the statechart diagram.

An analysis of the errors found in the testing phase leads to the following error classes:

1) About 70% of the errors found were logical errors and therefore design errors.

2) About 20% of the errors found were runtime errors, caused by forgotten initializations in the manually implemented code.

3) About 10% of the errors found were memory leaks.

The error classes two and three could be reduced in future projects to a minimum by refining the code generation properties. Class two errors will be reduced automatically by increasing the amount of automatically generated code (currently about 80 percent). The reduction of errors of class three can be done by the integration of constructor and destructor method implementations in the code generation templates. In the present templates only the standard constructors and destructors are generated automatically. Since the needed information is statically available in most cases improvements are possible for less effort. Only the first class errors cannot be addressed by increased use of automated code generation and require deeper further investigations. To reduce these errors the combination of the chosen approach with design level simulations and formal methods appears to be helpful. Another very promising circumstance is the fact that the work on the UML standard will go on. Influenced by the precise UML group (see e.g. [6]) the Object Management Group (OMG) plans the new standard UML 2.0 (see e.g. [7]). This will come with complete meta-modeling semantics. Concluding it can be said that the combination of UML as modeling language and Ada 95 as implementation language in conjunction with the ACD code generation leads to higher software quality. Our concrete example, Inspect 2, has been successfully running in industrial applications all over the world in round-the-clock operation for six month.

The future plans of our research group can be divided into different fields. One of these is the further development of

---

[5] Since our approach has a strong focus on the design phase, we use the term testing phase in an informal way.

Inspect 2 with the creation of new application modules for different application domains.

Another field ist the advancement of the present development approach. To increase the debugging capabilities, one actual task is the definition of a watch-dog-concept, which allows the tracing of transitions in a log file. Therefore the code generation templates are modified.

A further, long-term activity is the development of formal methods for the quality assurance of UML design level models. Actually, the investigation of the theoretical foundation of that work is in progress. That needs the definition of formal meta-modeling semantics of the UML profile. The main objectives of our formal methods are the proof of liveliness and timing properties on the design level.

Another very promising subject in the field of the UML-based analysis and development is test automation. For the definition of test cases sequence diagrams are well suited. The definition of templates for the ACD code generation seems to be a facile venture. Since our resources are restricted we cannot address this subject in our research group.

6 Acknowledgements

7 References

[1] C. Fouda, H.B. Keller, M. Kersten, J. Matthes, S. Zipser and T. Krakau (2002), *Systemhandbuch Inspect 2.1*, technical report, Forschungszentrum Karlsruhe.

[2] W.D. Heker (2002), *Generating Ada95 with StP/UML*, technical report, Aonix GmbH, available via stp-support@aonix.de or StP 8.3.

[3] M. Kersten and H.B. Keller (2001), Die Problematik der Abbildung von UML-Modellen auf Konstrukte der Programmiersprache Ada 95, in: P. Dencker et.al., editors, *Ada und Softwarequalität: Ada Deutschland Tagung 2001*, München, Ottobrunn, Shaker Verlag.

[4] S. Brodsky, T. Clark, S. Cook, A. Evans, and S. Kent (2000), *Feasibility Study in Rearchitecting UML as a Family of Languages using a Precise OO Meta-Modelling Approach*, technical report, The precise UML Group, http://www.cs.york.ac.uk/puml/mmf.

[5] OMG (1999), *OMG: Unified Modeling Language Specification*, technical report, OMG, http://www.omg.org.

[6] Clark et.al. (2001), *Initial Submission to OMG RFP's : ad/00-01-01(UML 2.0 Infrastructure), ad/00-09-03(UML 2.0 OCL)*, technical report, Precise UML Group, http://www.cs.york.ac.uk/puml/papers/uml2submission.pdf.

[7] OMG (2001), *Request for Proposal: OMG Document: ad/00-09-01 UML 2.0 Infrastructure*, technical report, OMG, http://www.omg.org.

Dr. Hubert B. Keller is head of the research groups "Innovative Process Control" and "Intelligent Sensor Systems" at the Institute for Applied Computer Science, Research Centre Karlsruhe.
Research interests: real time systems, software engineering, machine intelligence, intelligent sensor and process control systems.

Dipl.-Inf. Michael Kersten was part of the research group "Innovative Process Control" at the Institute for Applied Computer Science, Research Centre Karlsruhe from 1999 to 2002. Now he is working at the department of computer science at the Carl von Ossietzky University Oldenburg.
Research interests: design and analysis of object-oriented embedded real-time systems.
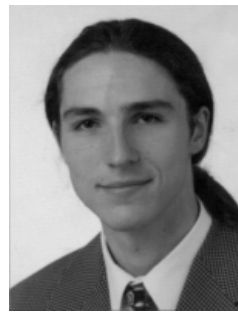
Dipl.-Ing. Jörg Matthes is part of the research group "Intelligent Sensor Systems" at the Institute for Applied Computer Science, Research Centre Karlsruhe since 2000. Research interests concern data analysis in sensor networks.

Dipl.-Ing. Christian Fouda Manga is part of the research group "Intelligent Sensor Systems" at the Institute for Applied Computer Science, Research Centre Karlsruhe since 2000. Research interests concern safe and secure interlinking of fieldbuses to Ethernet.

Dipl.-Ing. Stephan Zipser is part of the research group "Innovative Process Control" at the Institute for Applied Computer Science, Research Centre Karlsruhe since 2000. Research interests concern optimization of industrial combustion processes.

Email: {keller,matthes,fouda,zipser}@iai.fzk.de
        Michael.Kersten@informatik.uni-oldenburg.de

Ada-Europe 2002 Sponsors

ACT Europe
Contact: Zépur Blot

8 Rue de Milan, F-75009 Paris, France
Tel: +33-1-49-70-67-16 Fax: +33-1-49-70-05-52
Email: sales@act-eurpoe.fr URL: www.act-europe.fr

Aonix GmbH
Contact: Peter Dencker

Durlacher Allee 95, D-76137 Karlsruhe, Germany
Tel: +49-721-98653-22 Fax: +49-721-98653-98
Email: dencker@aonix.de URL: www.aonix.com

DDC International A/S (DDC-I A/S)
Contact: Erik Jensen

Gammel Lundtoftevej 1B, DK-2800 Lyngby, Denmark
Tel: +45-45-87-11-44 Fax: +45-45-87-22-17
Email: eje@ddci.dk URL: www.ddci.com

Green Hills Software GmbH
Contact: Hans-Joachim Fischer

Unterreut 6, D-76135 Karlsruhe, Germany
Tel: +49-721-9862580 Fax: +49-721-9862581
Email: sales-ger@ghs.com URL: www.ghs.com

RainCode Corp
Contact: Deborah Torrekens

1 Rue de l'Autonomie, B-1070 Bruxelles, Belgique
Tel: +32-2-522-06-63 Fax: +32-2-522-09-30
Email: Deborah@raincode.com URL: www.raincode.com

Rational Software Austria GmbH
Contact: Martina Rumpeltes

Lassallestraße 7b, A-1020 Wien, Austria
Tel: +43-1-245-31-160 Fax: +43-1-245-31-200
Email: info@rational.com URL: www.rational.com

TNI Europe Limited
Contact: Pam Flood

58A Mill Street, Congleton, Cheshire CW12 1AG, UK
Tel: +44-1260-29-14-49 Fax: +44-1260-29-14-49
Email: info@tni-europe.com URL: www.tni-europe.com