AdaCore
The GNAT Pro Company

Open Source
Open Systems
Free Software
and
Ada 2005/AdaCore

Robert Dewar
Emeritus Professor, New York University
President/CEO AdaCore

Sao Paulo, May 6th, 2009

Open Source

- Open Source is a relatively new term
- It refers to the notion of freely providing sources and encouraging a wide community to participate in development
- Attractive to major companies (gee, maybe we can get people to do stuff free for us ☺)
- Claims are made for better quality, better security etc.
- In practice, some OS projects work, some don’t. Some OS software is high quality, some is not.

Free Software

- A term coined a long time ago by Richard Stallman of the Free Software Foundation.
- Refers to the notion of ensuring that the user/recipient of software has freedom to do what they like with it.
- Free = Free as in freedom, not free as in free lunch (libre, not gratuit).
- The emphasis is on this freedom for users.

Open Systems

- An Open System is one built from components with well defined open interfaces.
- Free from patent or copyright encumbered formats and interfaces (e.g. no wmf files)
- No secret interfaces (e.g. NT kernel)
- The idea is that anyone can plug in anything at any point, by adhering to these interfaces.

Open Systems, Reality Check

- **A real open system is one that adheres to the philosophy of openness throughout. For example GNU/Linux systems.**
- **Just because you can pass POSIX tests does not mean you are open (e.g. POSIX subsystem for NT)**
- **Just because you have some open components does not make a real open system (e.g. MAC OS X).**

Who is AdaCore? Company Overview

- **Engineering organization focused on Ada**
- **Largest team of Ada experts**
- **Committed to Open Source**
- **Worldwide presence**

NewYork and Paris HQ

Branch offices in
California, Sweden,
Japan and Australia

Domains and Customers

Application Domains

- Air traffic control
- Avionic (civil & military)
- Communications
- Electronics
- Energy
- Financial institutions
- Medical imagery
- Military (airborne, ground, naval)
- Space
- Telecom
- Television
- Transportation

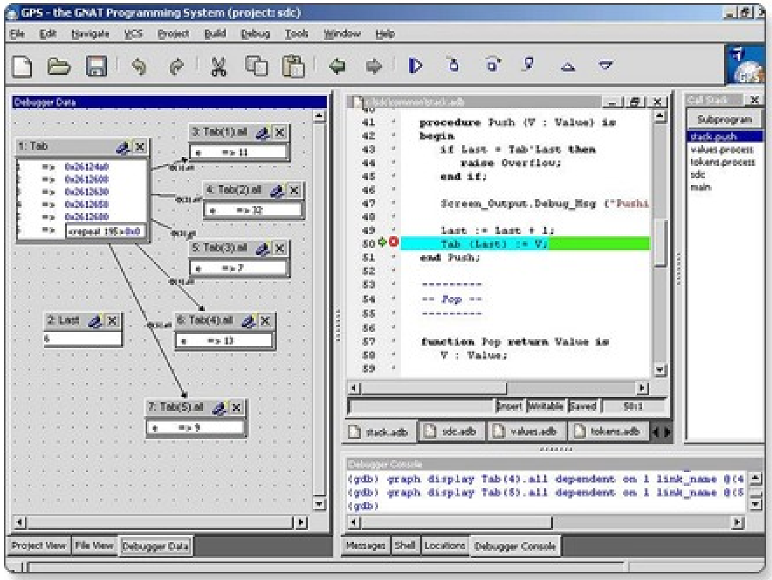
Customer s

- Alcatel
- Alenia
- Alstom Transport
- BAe
- Boeing
- EADS
- Embraer
- Ericsson
- Eurocontrol
- JEOL
- Lockheed Martin
- MBDA
- Nagra
- Philips Semiconductor
- Raytheon
- Rockwell Collins
- SAAB
- Smith Industries
- THALES

Ada Software Development Solutions

- **GNAT Pro**

Full-featured Ada 95 development environment



```
procedure Push (V : Value) is
begin
   if Last = Tab'Last then
      raise Overflow;
   end if;

   Screen_Output.Debug_Msg ("Pushi

   Last := Last + 1;
   Tab (Last) := V;
end Push;

---------
-- Pop --
---------

function Pop return Value is
   V : Value;
```

```
(gdb) graph display Tab(4).all dependent on 1 link_name @(4
(gdb) graph display Tab(5).all dependent on 1 link_name @(5
(gdb)
```

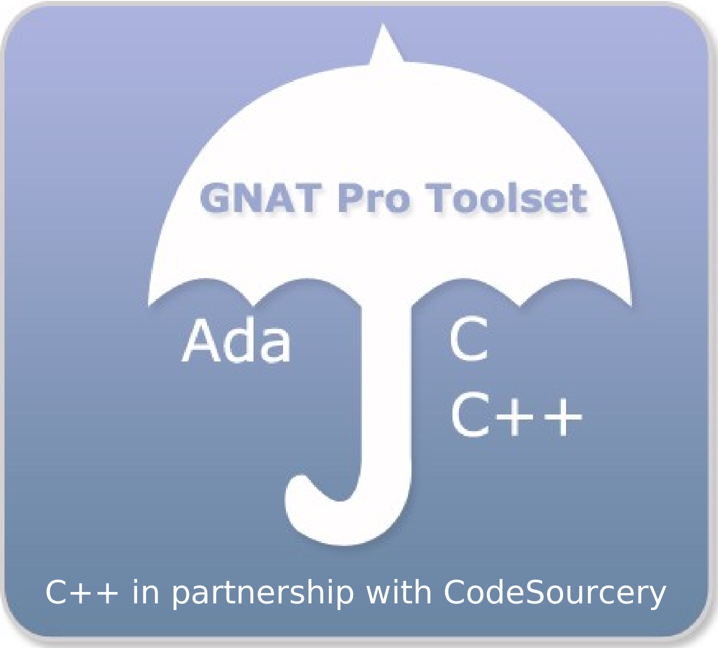
Ada Software Development Solutions

- **GNAT Pro High-integrity Editions**

  For developing Safety Critical Applications in Ada

Mixed Language Solutions

- GNAT Pro offers **Ada, C, C++ interoperability**



Domain-Specific Solutions

- **GtkAda**
  - To develop modern, native GUIs for UNIX and Windows in Ada
- **ASIS-for-GNAT**
  - To develop portable tools for Ada software

Distributed Systems Solutions

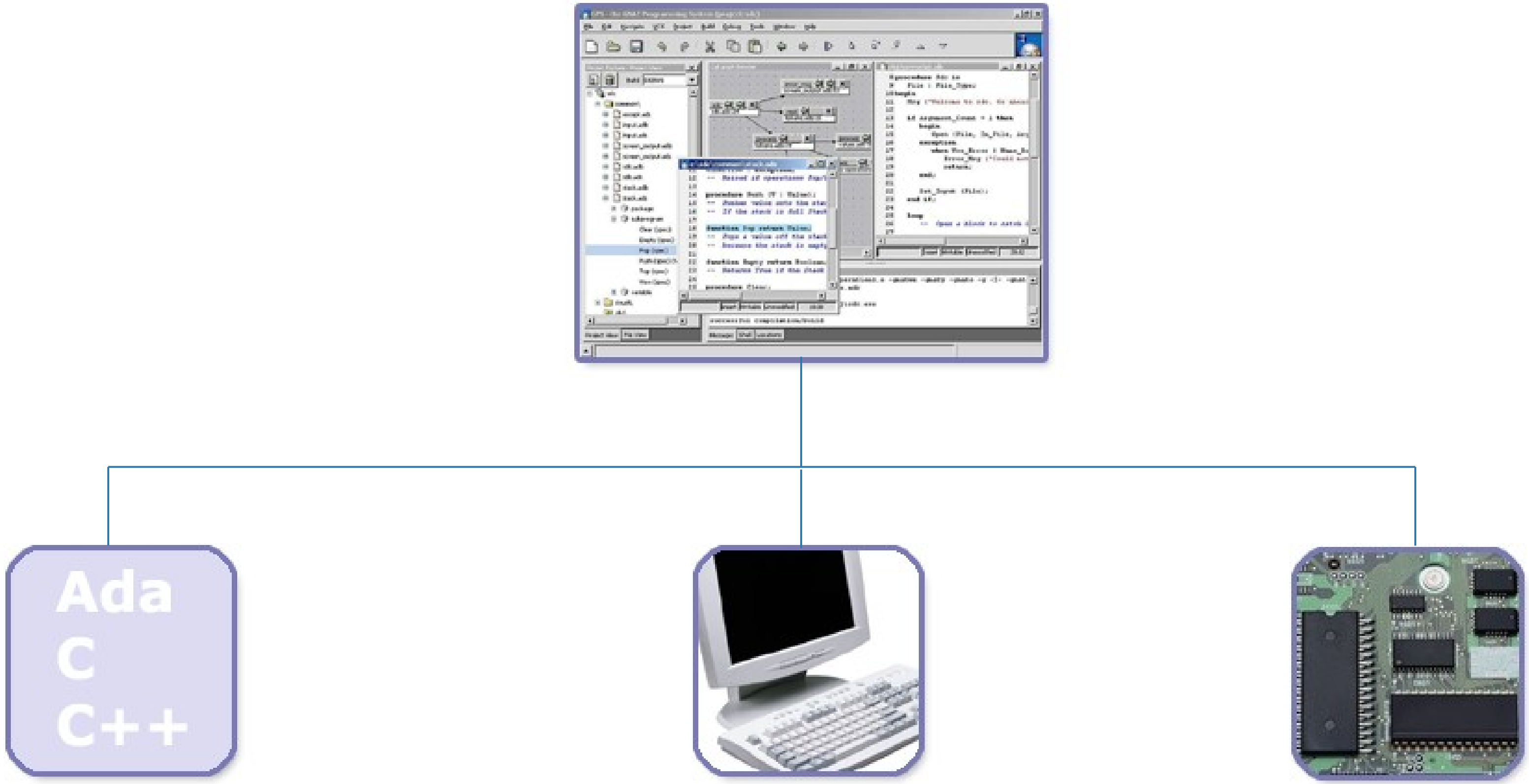
- **GLADE**
  - To build distributed systems using the Ada 95 Distributed Systems Annex
- **PolyOrb**
  - To build distributed systems using CORBA and Ada

Web-Related Solutions

- **AWS**
  - To web-enable Ada applications (web server development environment)
- **XML/Ada**
  - To process XML streams in Ada applications

GNAT Pro Key Point

A common technology in a single IDE



Mixed-Language Development

Native/Host Development

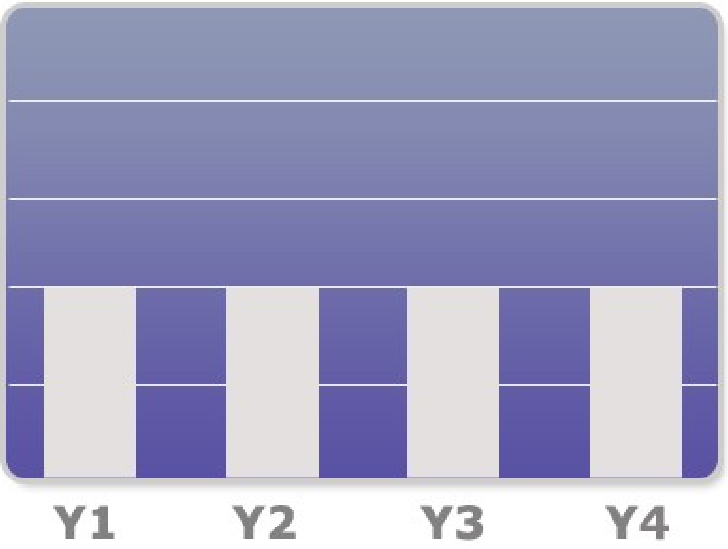
Embedded Development

Support: Aligned with Your Interests

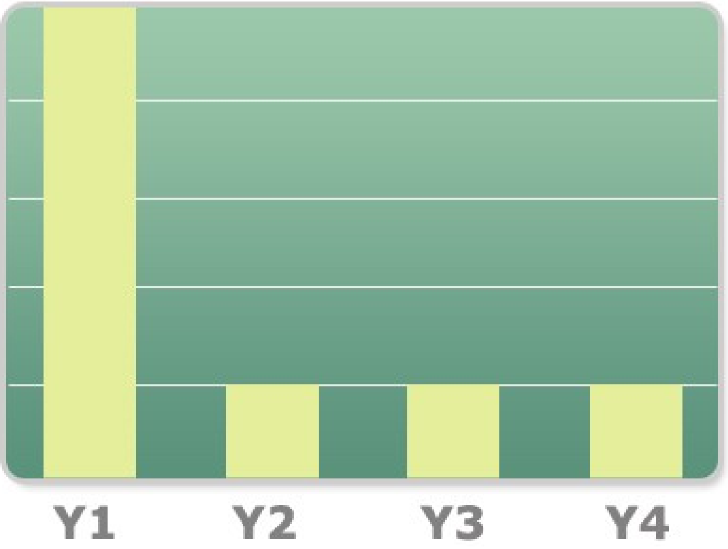
- Our development solutions are sold as a subscription
- Our development solutions are open source
- Direct access to AdaCore experts

Our Yearly Subscription Model

AdaCore Subscription Model



PUF - Pay up front Model



Advantages

- Less financial risk
- Consistent quality of support every year

The Open Source Advantage

- **Source code included**
- **Lock-free**
- **All developers can use it at any time**

A Comprehensive Solution

The GNAT Pro subscription package includes:

Powerful Tool Suite

+

Support & Online Consulting

Tool Suite Support

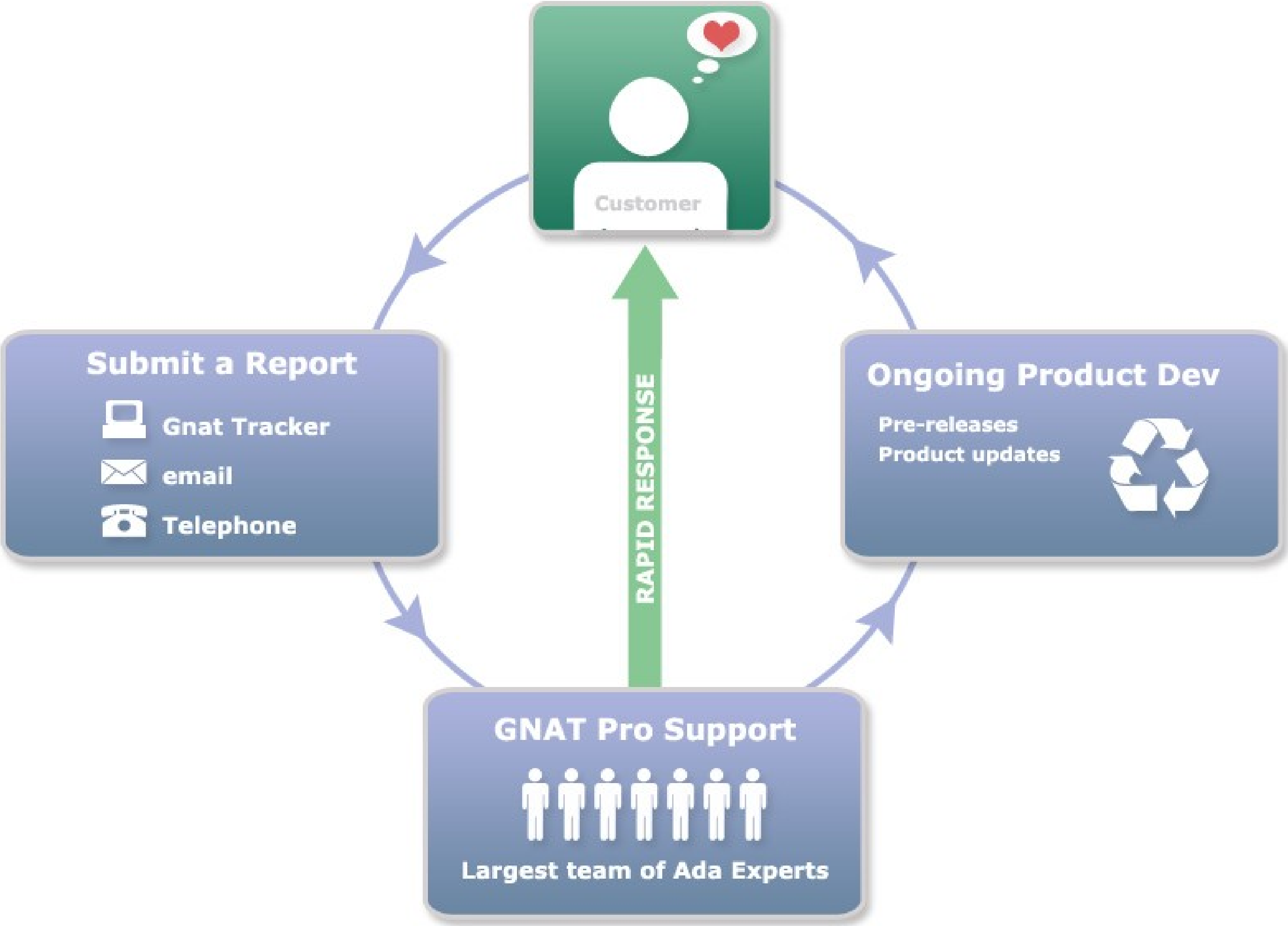
- Help with installation procedures
- Explanations regarding implementation dependant features
- Correction of problems encountered in the use of tools
- Workaround provided if a problem is encountered
- Pre-releases provided for high priority compiler problems

Online Consulting

- Expert answers to questions about Ada 83 and Ada95
- Suggestions for the effective use of Ada 95 in your project
- Advice on ways to structure your project using GNAT Pro
- Suggestions on best use of GNAT tools for your project
- Suggestions on how to migrate from other Ada technologies
- Help with all aspects of Ada 95 software development

Support + Online Consulting: How it Works

Support + Online Consulting: How it Works



Customer

RAPID RESPONSE

Submit a Report
- Gnat Tracker
- email
- Telephone

Ongoing Product Dev
- Pre-releases
- Product updates

GNAT Pro Support

Largest team of Ada Experts

GNAT Tracker - Customer Web Server

- Secure Access
- Easy download of GNAT Pro tool suite and components
- Complete online version of GNAT Pro documentation
- Submit toolset support or online consulting requests
- Easy download of GNAT Pro releases
- Access of up-to-date versions of known-issues and features
- Browse/search all request exchanges for your account

Additional Services

- **GNAT consulting**
- **Training**
  - Ada95
  - GNAT technology
- **Professional services**
  - GNAT Pro ports
  - Special-purpose runtimes
  - Tool customization & enhancements
  - Assistance with in-house specialized GCC & GNAT ports

Free Software vs Open Source

- **One of the important freedoms for Free Software is the freedom to modify, which means that sources are available.**
- **So it is often, but not always, the case that FS ends up with an open source community participating in development.**
- **But not all OS projects are free software because licenses may be too restrictive.**

OS vs FS. What’s Important to You?

- **Open Source is not an important issue for the user. In fact there are plusses and minuses in open source:**
  - Pluses
    - May be higher quality due to large developer community
    - May be more secure due to many eyes
  - Minuses
    - Large open developer communites are hard to control. Out of control development is not good.
    - Quality control may not be a major focus of a community of enthusiastic developers.

OS vs FS: What’s Important for You?

- **Free Software**
  - The emphasis is on liberal license terms
  - That means that you as a user can do more and have fewer restrictions
  - That’s an unqualified plus
- **What about quality?**
  - FS does not guarantee quality
  - You will have to use your normal procedures to make sure that you choose quality software!

Free Software Licensing

- Microsoft sells exclusively proprietary programs with a restrictive license
- AdaCore sells programs covered by a Free Software License (GPL and GMGPL)
- So that’s completely different, right?
- Actually, there is much more similarity than you might think.

Free vs Proprietary Software

- **Microsoft and AdaCore both:**
  - Sell software with support
  - The software is copyrighted, and cannot be copied or used without a license.
  - Provide a license that allows you limited rights to copy the software and use it.
  - You pay for the combination of the software, support, and appropriate license.
- **So what is the difference?**

It’s All in the License Terms

- The ONLY difference between MS and AdaCore is the terms of the license
- But it’s quite a difference because the license conditions are very different.
- The MS license is all about strictly limiting the usage and copying.
- The AdaCore license is far more liberal and has very few restrictions.

Differences

- **Anything the MS license allows is also allowed by the AdaCore License**
- **But the AdaCore (GPL/GMGPL) is much more permissive**
  - Unlimited copying allowed
  - Unlimited use
  - Unlimited distribution
  - Full sources available
  - You can modify and distribute sources
  - Run-time library can be modified and used freely in any of your programs in any way you like.

So What Can’t I Do

- The GPL/GMGPL allows a lot but it does stop you from doing some things.
- You can modify the software (e.g. the Ada 95 compiler, GNAT) and use it freely
- You can even distribute the modified version.
- But if you do, you must use the GPL for that distribution and include sources including your modifications.

GPL: A Virus?

- **I have heard that the GPL is a virus which will cause me to lose all my rights to my own code.**
- **Yes, there are some parties that have been energetically spreading this peculiar rumor.**
- **Let’s look at the facts and you decide for yourself if Virus is an appropriate term ☺**

The GPL Virus in Action ???

- **Suppose you:**
  - Obtain the sources
  - Modify them
  - Distribute the modified version
- **What do the two licenses have to say about this sequence of events?**

The Microsoft Response

- **Obtain the sources**
  - Where did you get them? Our lawyers want to know. That’s stolen property, cease and desist or else.
- **Modify them**
  - You are creating an illegal deriviative work. Enclosed is a notice of a suit we are filing for copyright infringement. You have no rights to do this.
- **Distribute the modified version**
  - This is getting more serious, the statutory penalty for such distributions, which violate our copyright, is $50,000 for each distributed copy.

The AdaCore Response

- **Obtain the sources**
  - We hope you didn’t have any trouble locating the sources. Let us know if you did and we will help you out. The documentation contains full details on how to obtain and build them
- **Modify them**
  - You are welcome to do so. We can provide some help under your standard support contract, but if you make major modifications, then this may limit the support we can provide. We are happy to discuss special support arrangements. Also you might want to discuss your modifications with us. Perhaps it is something we would be happy to do. Perhaps you would like to contribute your modifications so they will be in future releases.

The AdaCore Response, continued

- **Distribute the modified version**
  - No problem. The only rule is that you need to distribute the full sources (our original stuff with your modifications), and use the same license (the GPL) for everything. You don’t have to distribute, and as long as you don’t you are welcome to keep your modifications secret.

The Virus Effect, Part 2

- **OK, I understand about modifying the compiler, and that’s fine, I’m not in the compiler modification business anyway.**
- **But what about the run time, don’t my programs include some of your run-time?**
- **Answer: yes, they do. If the run-time was under the GPL, there would be a problem.**

A GPL’ed Library or Run-Time

- Even if you don’t modify, your program that includes such a library would be covered by the GPL.
- That’s fine for people who want to write Free Software
- But not so fine at all for people who want to write non-free software, e.g. classified weapons system software.

Worrying about Licenses

- **Free Software and Proprietary Software share an important common truth:**
- **CHECK THE LICENSE!**
- **Make sure it is suitable for your use**
- **A GPL’ed library (e.g. CYGWIN from Redhat) is not suitable for use in non-free software.**
- **So you need to look elsewhere (e.g. Redhat sells a separate version of CYGWIN with different more suitable licensing).**

What About the AdaCore License?

- We know our compiler and tools are going to be used in non-free-software contexts.
- After all a great deal of the Ada business is in defense systems for example.
- So we make very sure that our licensing is appropriate for this use.
- And we provide a simple and clear license statement as part of the product that guarantees that this is the case.

The GPL and the GMGPL

- **The GMGPL is used in GNAT for ALL components which might be included in your program:**
  - Run Time Library
  - General use libraries
  - Generic code you need to instantiation
- **The GMGPL specifically allows you to incorporate this in your program without any effect on the IPR status. The program can be proprietary, classified etc completely at your discretion.**
- **CHECK THE LICENSE!**

The GPL Version of GNAT

- **The GPL version includes all our technology**
- **But under the pure GPL**
- **Just fine for use in**
  - Universities
  - Hobbyists
  - Experimenting
  - Writing free software
- **But not fine for**
  - Writing avionics code for the Boeing 787!
- **GPL version used by our GAP university program**
  - Free access to the GPL technology with full support from us
  - Cost: free (this is free free software!)
- **CHECK THE LICENSE!**

COTS and Free Software

- **Everyone agrees COTS is a great idea**
  - Reduced Costs
  - Standard software reduces training costs
  - Economies of scale
  - Often better design from greater resources
  - Inexpensive way to stay with the state of the art in technology.

So What’s the Downside with COTS?

- **Two big downsides**
  - Vendor lock in for support
    - Only the vendor can provide support
    - This can be locked in with licenses etc
    - If the vendor goes bankrupt, too bad
    - Source escrows are not much help
  - Vendor lock in for modifications
    - If the software does almost what you want, but not quite, you have to ask the vendor for changes.
    - This can be arbitrarily expensive!

Free Software to the Rescue!

- **Fixing the two big downsides of COTS**
  - Vendor lock in for support
    - Everyone has access to the sources
    - Anyone can provide support
    - You can even build your own support
    - If there is a demand other companies will compete
  - Vendor lock in for modifications
    - Everyone has access to the sources
    - Anyone can do modifications
    - You can do modifications yourself if you like
- **COTS + FS = COTS without the risks!**

So How Does AdaCore Make Money?

- **It’s probably the misinterpretation of the Free in Free Software that generates this rather peculiar question?**
- **The answer is pretty simple:**
  - We provide a good product at a good price ☺
  - If the quality is bad, we fail ☹
  - If the price is too high, we fail ☹
  - Which is what capitalism is all about!
  - We welcome competition!
  - You could even compete with us doing GNAT maintenance!

What Product Does AdaCore Provide?

- **An excellent Ada 95 compiler and development system, with a full set of tools, IDE, libraries, etc. Also support for use of C and C++ used with Ada.**
- **Excellent support (high quality online consulting services)**
- **Constant improvement and upgrades**

But Why Would I Buy This Product?

- **I can download the latest version (free!) and build it myself.**
  - That’s almost true, putting everything together correctly is not trivial, and some ports are quite tricky to do.
  - You can’t get the exact set of sources that correspond to our release
  - Also, you need to take responsibility for all licensing and IPR issues when you obtain anything from the net.
  - It will take you more time than you think.
  - This version will not be tested
  - The tests we have (tens of millions of lines) are proprietary
  - You are paying for our testing and support

Why Would I Buy This Product?

- **OK, So perhaps I won’t build it myself, but there are public versions around already built that I can use.**
  - Yes, there are. In fact we try to make sure this is the case for commonly used targets student/research etc.
  - But again, you need to be careful about what you download.
  - And you don’t get any assurances of quality
  - Many bad versions of GNAT have been distributed. Well not so bad for student use, but not suitable for mission critical use.
  - You won’t find specialized versions (e.g. high integrity versions with certification materials, or AE653 builds etc).
  - And of course these versions are not supported

Why Would I Buy Your Product?

- **You mentioned support, but I know what I am doing, and I can do fine without support.**
  - OK, if you are sure that is the case, fine, but a compiler with a full set of tools requires a lot of expertise.
  - And do remember that in this case you take responsibility for issues of proper licensing and respect for intellectual property.
  - Our customers find that the support saves them money and time!

The AdaCore Business Model

- We need to provide an excellent product with excellent support.
- Our mission: Make sure you renew support. Unlike Microsoft, we can’t do this with restrictive license agreements, so we have to do it with high quality ☺
- We need to be constantly improving and extending our product and support so you continue to benefit from our services.
- Not so different from any other business!

How Are We (AdaCore) Doing?

- We are doing fine!
- We have over 50 full time engineers and about 8 sales/ administrative people.
- We have run from revenues from the start with no outside investment
- We celebrate our tenth anniversary in July of this year. We are consistently profitable and growing steadily and gradually.
- We won’t get rich like Bill Gates, but we pay ourselves decent salaries (no engineer has ever left Ada Core Technologies ☺)
- We all enjoy our jobs, the combination of challenging and interesting software development, along with directly supporting our customers and their projects makes a great environment. We also appreciate that the openness of our sources means everyone can read what we write ☺

What Can We Do For You?

- We can offer one of the most advanced and capable compilers for any language available, together with a highly advanced IDE.
- Ada remains an excellent choice for critical software. The design criteria for Ada remain valid today, and indeed, in these days of ever more critical software applications, the need for languages that make it easier to produce reliable software is greater than ever.
- Particularly in the realm of safety critical software, Ada has been very successful, and that’s not surprising. The focus in Ada on safety is fundamental. You won’t find a specific section in any other language standard on safety and security.
- We offer excellent support. In fact this level of support, is another reason for choosing Ada.

Can This Model Work for All Software?

- **Probably not**
- **For example, computer games**
  - Really don’t need support
  - Don’t need new versions and new features
  - Can cost over 50 million US dollars
  - Need copyright protection here I think
  - More like movies than high level software
- **But for many other cases most likely yes**
- **How about Microsoft Windows**
  - Support is needed (e.g. security updates)
  - New versions and features
  - Probably people would pay for the “real thing”
- **Other examples (from our world)**
  - SPARK (system and language and tools for formal methods)
  - CodePeer (high level static semantic tool)