



# FLOSS and Business

---

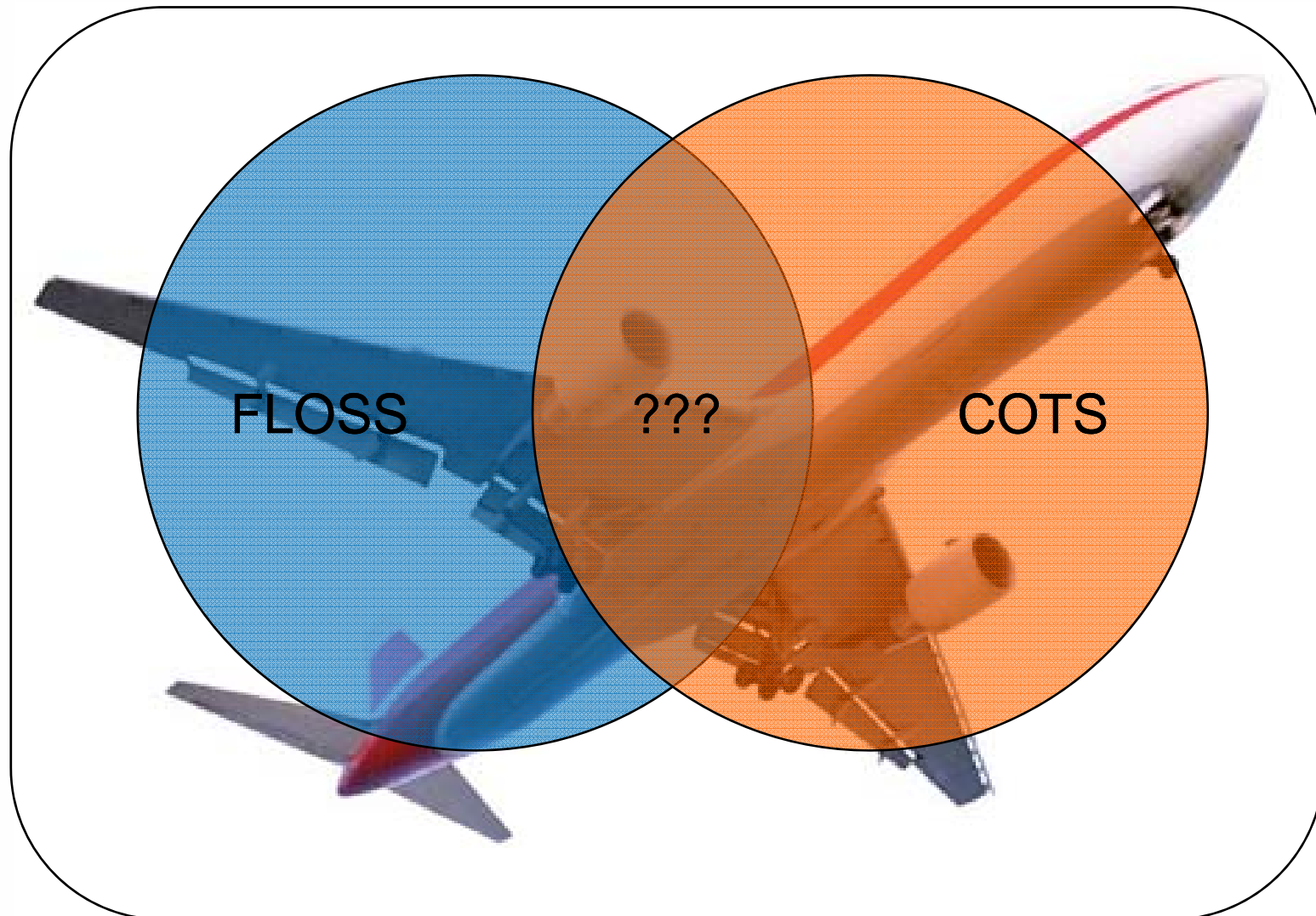
... or how to make money with FLOSS

Franco Gasperoni

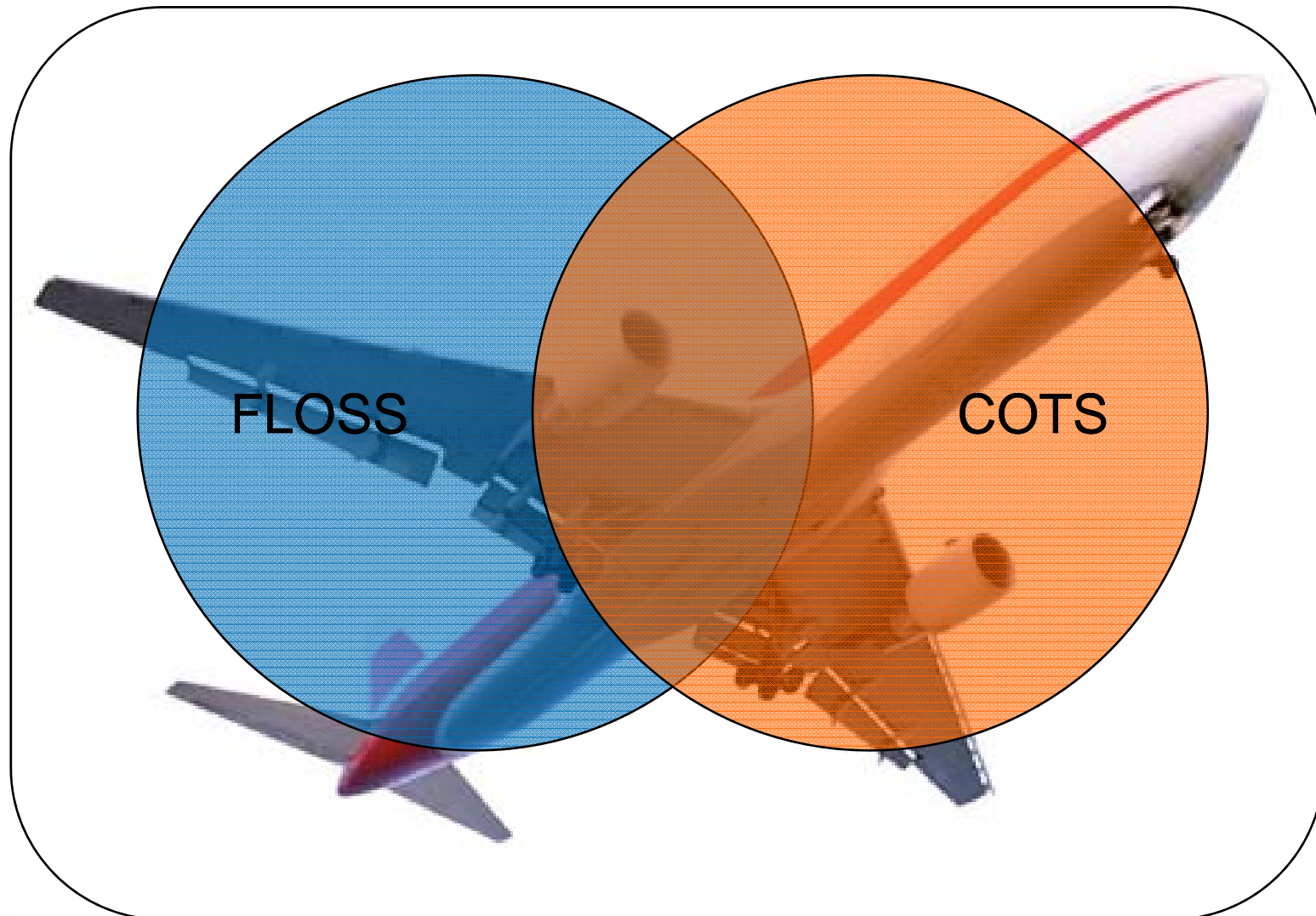
AdaCore

[gasperoni@adacore.com](mailto:gasperoni@adacore.com)

## Focus of This Talk



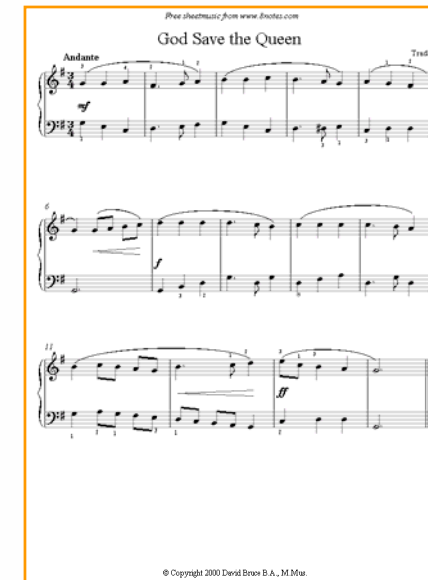
## Legal Background



## Legal Background: Copyright

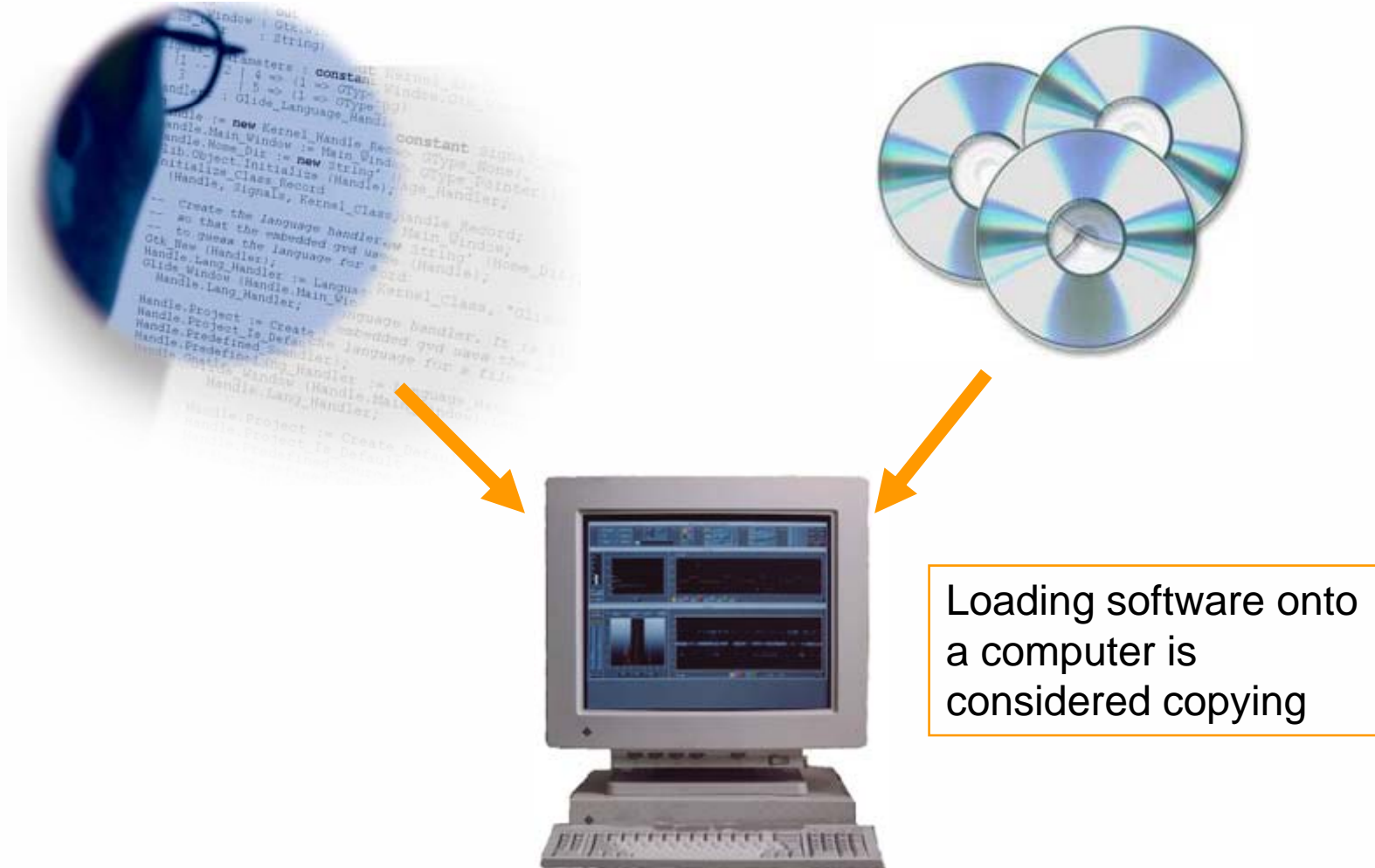
Exclusive legal right of copyright holder to

- Copy
- Distribute
- Modify
- Display
- Perform
- Exploit a work



## Software and Copyright

Both source and object code can be copyrighted



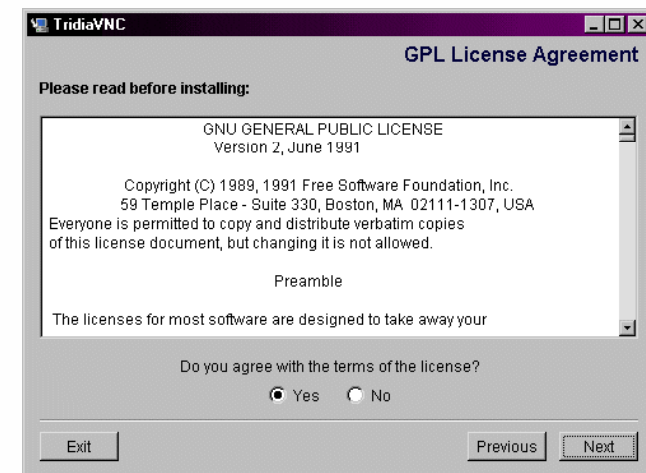
# Copyright & Licensing

A copyrighted work cannot be copied unless...

... the copyright holder grants you a license ...

... permitting copies under specified circumstances

**Virtually all software today is sold with a license**



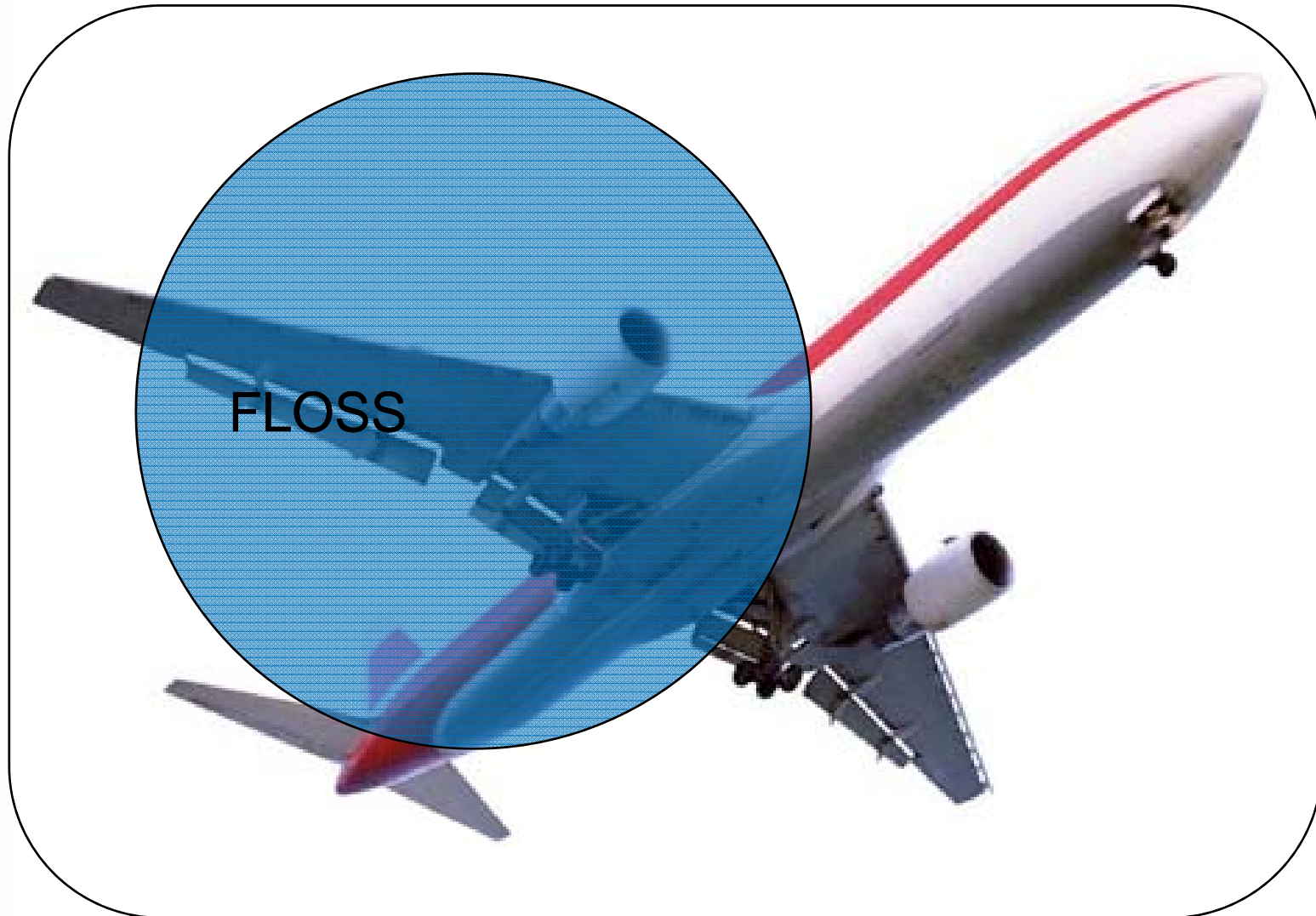
## *Free Software Is Copyrighted Software*

To protect these freedoms *Free Software* comes with a license

- The General Public License (GPL)

From this point there is **NO DIFFERENCE** between





# The Free Software Movement

Term coined in 80s by Richard Stallman

- *Freedom* to run
- *Freedom* to redistribute copies
- *Freedom* to study and adapt
- *Freedom* to improve the software and release improvements

Access to the sources is a precondition for the last 2 items

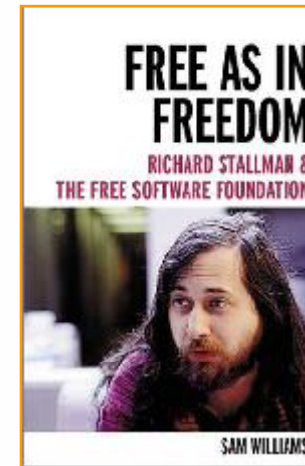
Emphasis on User Freedom



## The Meaning of *Free* in FS

### *Freedom*

- You can sell it
- You can make it available for free



**FS is a matter of liberty not price**

## Open Source Software (OSS)

---

- **Providing the sources** (*under some license*)

+

- **Encouraging a wide community to participate in development**

**There have been abuses in the licenses used**

- **Open Source Initiative (OSI) created to**
  - Define what licenses qualify as “Open Source”

## The OSS Movement

---

- **Attractive to major companies (e.g. IBM, SGI, HP, ...)**
  - Can leverage on a larger developer community
- **Claims are made for better quality, better security etc.**
- **In practice:**
  - Some OSS projects work, some don't.
  - Some OSS software is high quality, some is not.
  - Some projects make sense as OSS some don't.

## FS and OSS

---

- **One of the important freedoms for FS 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.
  - E.g. Linux
- **Not all OSS projects are FS because of the license**

## FLOSS

---

- **F**reely **L**icensed **O**pen **S**ource **S**oftware

- Free Software

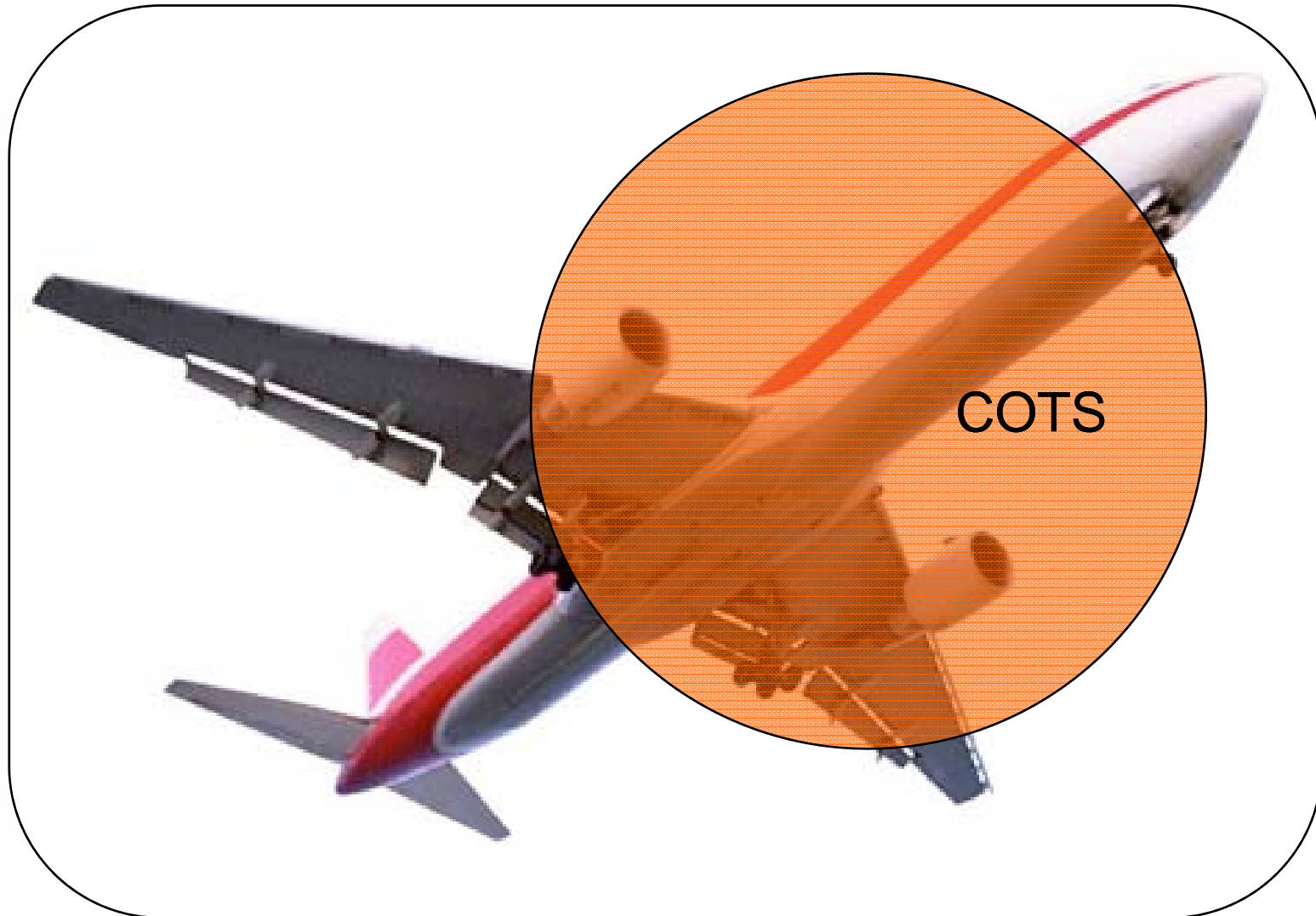
emphasis on user's right



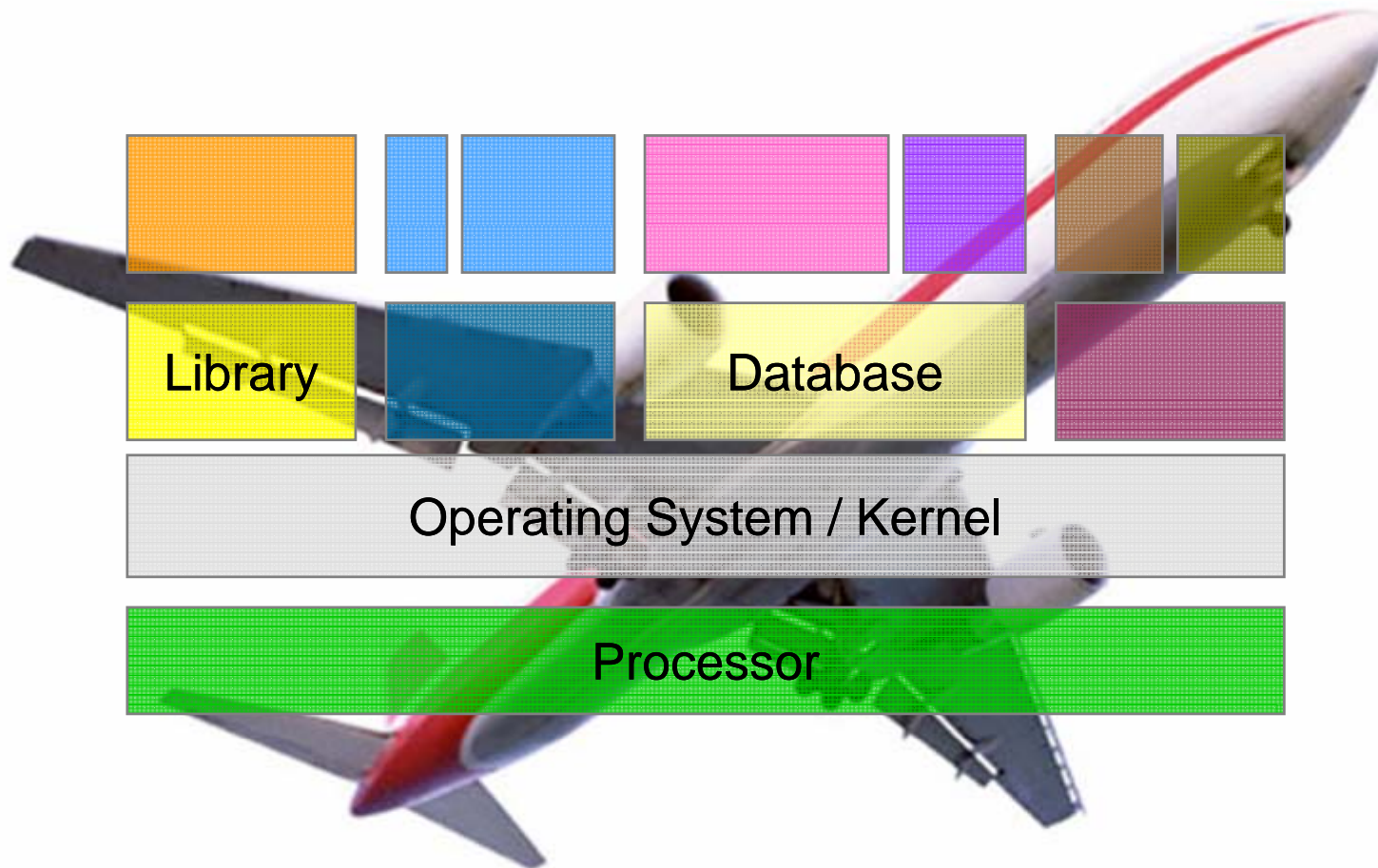
- Open Source

emphasis on open development process





## Software and the Market



Reuse?

Build?

Subcontract?

Buy COTS?

## What is COTS?

---

The market answer for **sharing cost** for the

**Creation**

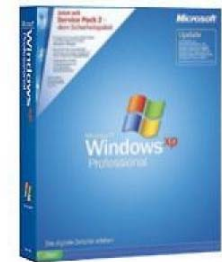
**Industrialization**

**IPR liabilities**

**Support**

**Maintenance**

**Repair**

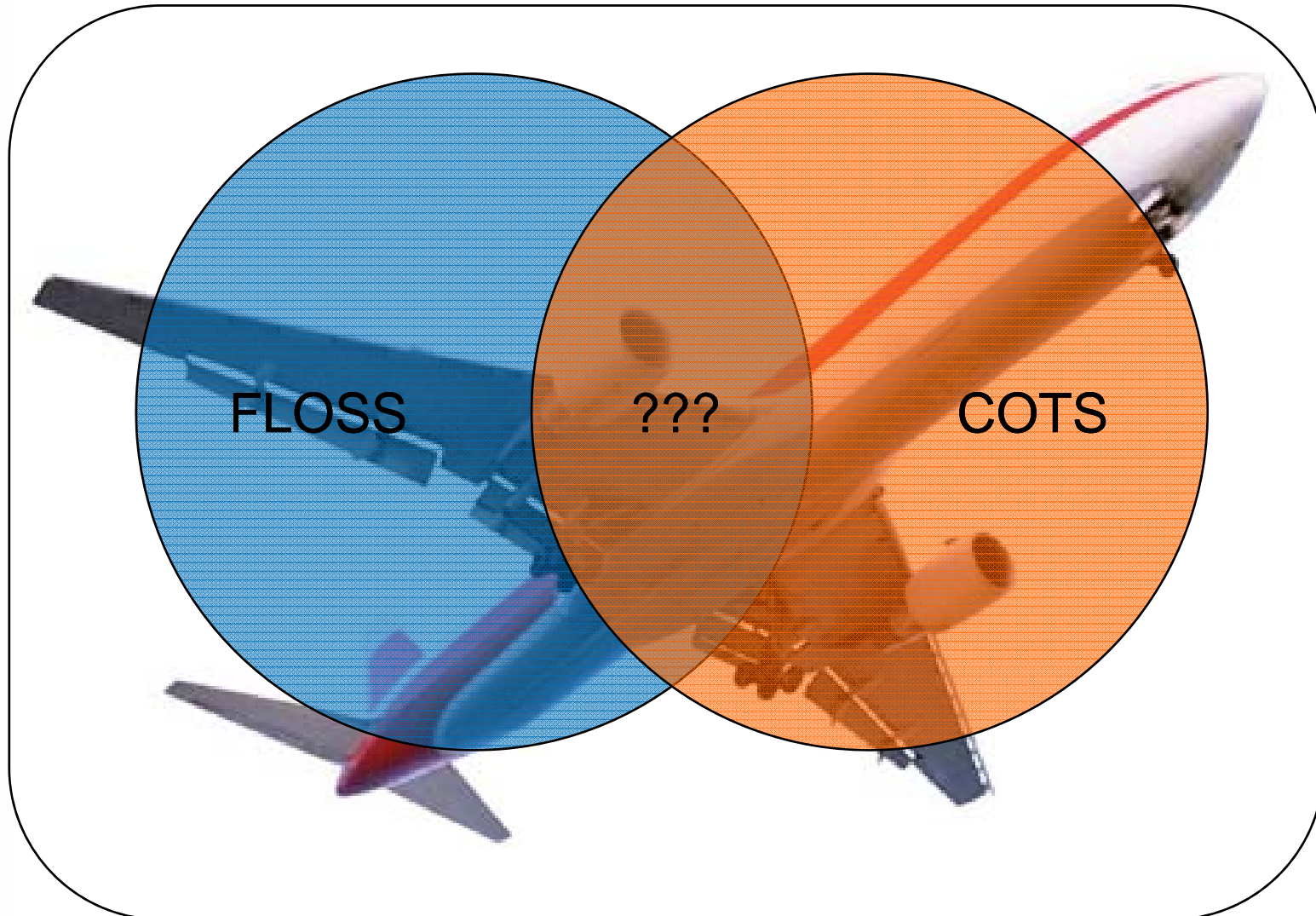


## The COTS Vendor

---

**What does a COTS vendor provide/sell?**

- 1. The COTS**
- 2. Liability for IPR issues**
- 2. Support over the entire COTS life-cycle**



## Two Types of COTS

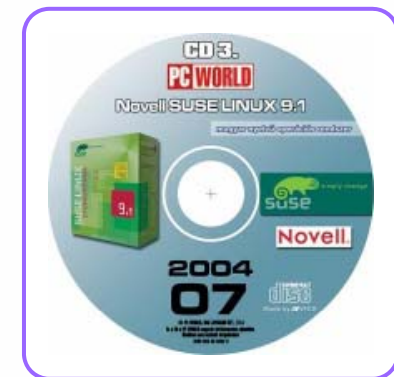
---

- **RL-COTS: Restrictively licensed COTS** (conventional)
  
- **FL-COTS: FLOSS licensed COTS** (AdaCore, Red Hat, ...)

## RL-COTS vs FL-COTS

- **COTS software**
  - The software is copyrighted
  - It cannot be copied or used without a license
  - The license allows certain rights to copy and use the software
  - Customers pay for the combination of: software, support, and license

**From this point of view there is NO DIFFERENCE**



## Restrictive vs FLOSS License

---

**The difference is all in the license terms**

- Big difference because license conditions are very different
- Restrictive license strictly limits usage and copying
- FLOSS license is far more liberal and has few restrictions

## Copyright and Monopoly (Vendor Lock-in)

---

- **Copyright CAN be used to establish:**
- **Monopoly for**
  - COTS Software customization
  - COTS Changes
  - COTS Evolution
- **Monopoly for support**
- **Monopoly for certification material**

## Duration of the Monopoly

---

- **Life + 70 years (humans)**
- **70 years after first publication (corporations)**
- **95 years in the US (corporations)**

## COTS Support

---

If unhappy with vendor support for:

Fixes

Customization

Updates

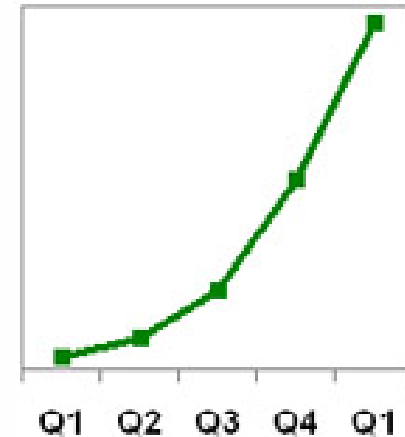
Maintenance

.....

What are your options?

## Switching COTS

Purchase  
Training  
Integrate into infrastructure  
Customize & adapt  
Use  
Fix problems  
Deploy  
Maintain for 20–40 years  
Reuse in next project



**Switching costs of COTS in safety-centric industries are high**

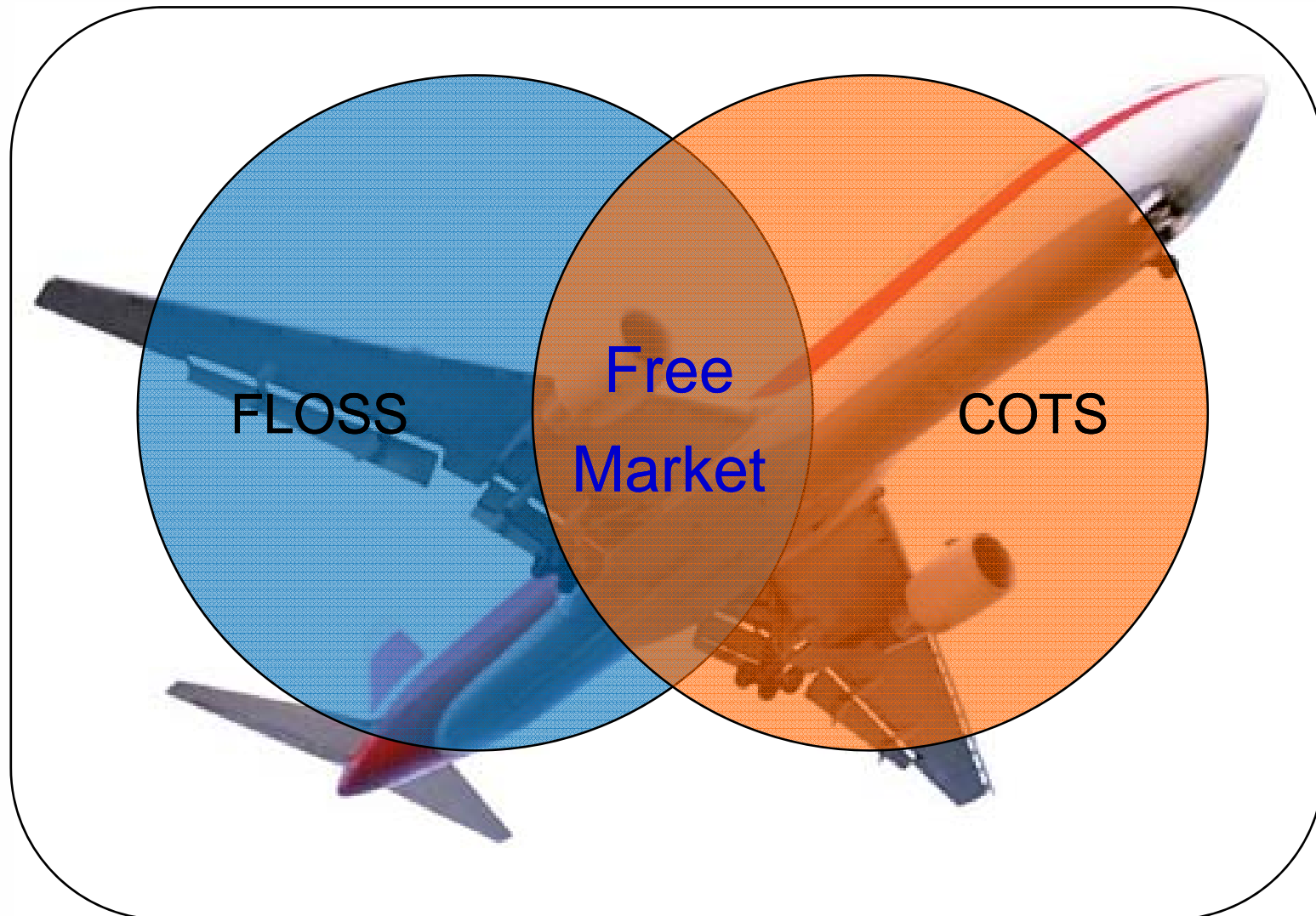
**Cannot switch if COTS has been deployed**

- life-cycle of 20 to 40 years

## RL-COTS and Monopoly

---

- **Monopoly for support, customization, and evolution**
  - Only the vendor can provide support
  - If the vendor goes bankrupt, too bad
  - Source escrows are not much help
- **Monopoly for customization & 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
- **Monopoly for certification material**
  - E.g. RTCA/EUROCAE DO-178B/ED-12B



## FL-COTS: Free as in Free Market

---

- **NO vendor lock in for support**
  - Customers have access to the sources & build procedures
  - Anyone can provide support
  - Customers can build their internal support team
  - If there is a demand other companies will compete
  
- **NO vendor lock in for modifications**
  - Customer has access to the sources & build procedures
  - Anyone can do modifications
  - Customers can do modifications if they like
  
- **NO vendor lock in for Certification material**

## FL-COTS and Market Freedom

	FL-COTS	RL-COTS
Switch COTS	YES	YES
Use other party for support	YES	NO
Use other party for changes	YES	NO
Use other party for certification material	YES	NO

## Quality Guarantees of RL-COTS & FL-COTS

- **NONE**
  - Unless evidence, such as certification material, is produced to the contrary
- **RL-COTS: May not even have access to sources**
- **FL-COTS: Even with access to sources/build procedures**
  - No license requires:
    - Distribution of design documents
    - Development plans
    - QA procedures
    - Test suites
    - Other software development artifacts
- **Nothing prevents FL-COTS from achieving high-quality**
  - and being used in safety-critical systems

## **FL-COTS: Freedom for the Entire COTS Life-Cycle**

---

- **NO monopoly on COTS changes**
- **NO monopoly for COTS support**
- **NO monopoly for COTS certification material**

**Aligns COTS vendor & customer interests around**

High-quality products and services

## Worrying about Licenses and Quality

---

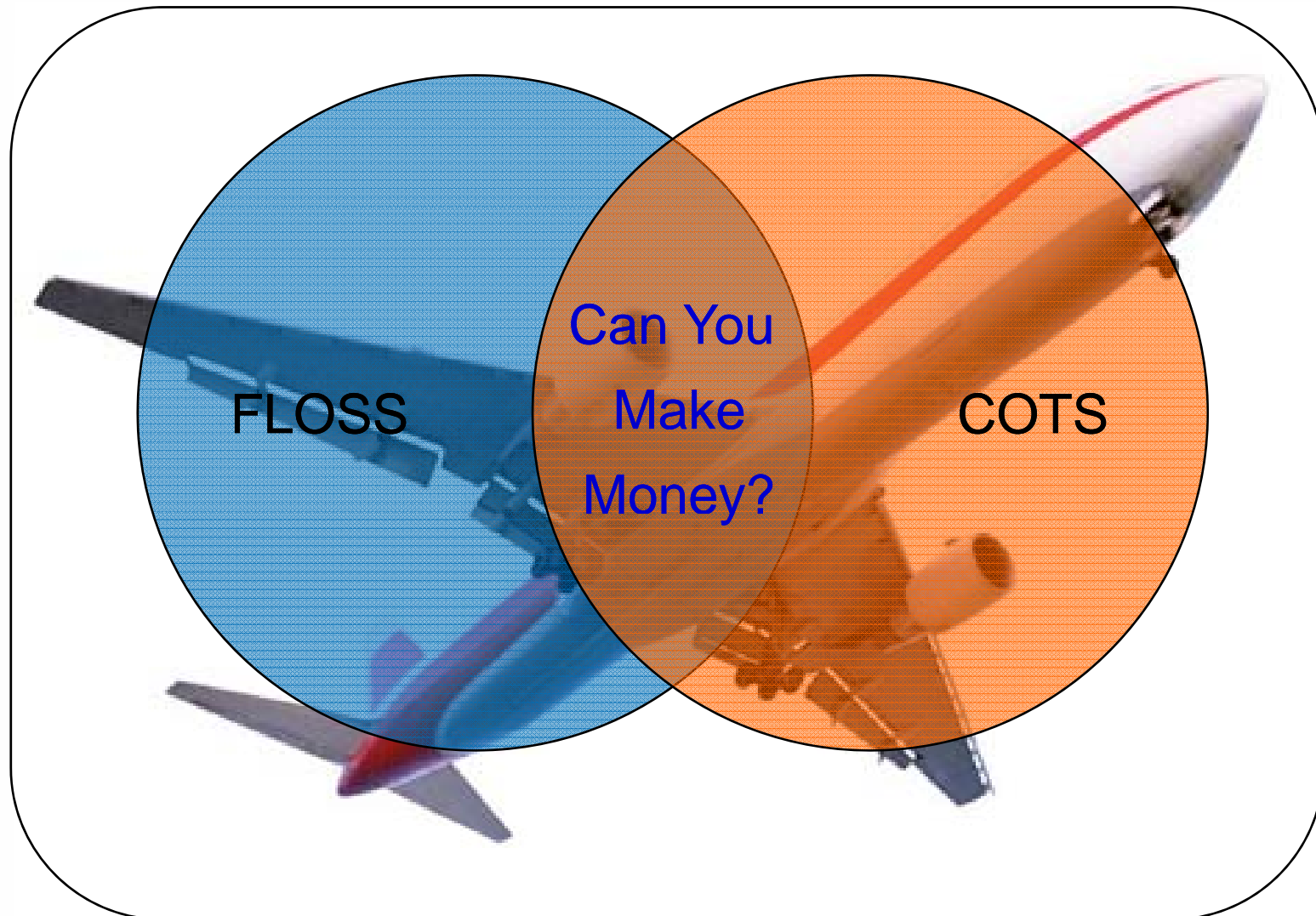
FS, OSS, and Proprietary Software share 3 common truths

- **CHECK THE LICENSE**
  - Make sure it is suitable for your use
  
- **CHECK THE QUALITY**
  - No software license guarantees quality
  - Use your normal procedures to ensure that you choose quality software
  - Buy SW products whose business model aligns with your quality needs

**FLOSS + COTS = 100% Free Market**

---

FLOSS is an unqualified plus  
for the COTS customer



## AdaCore

---

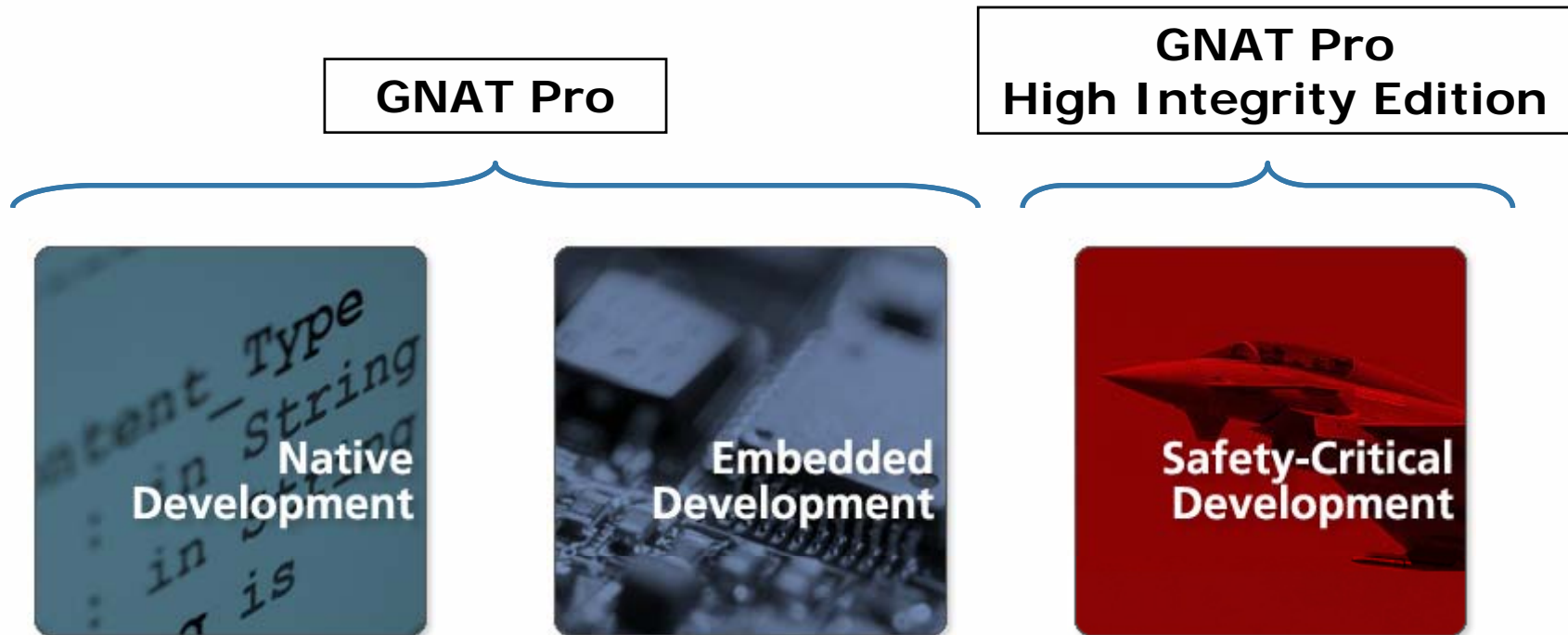
- **Engineering organization focused on Ada**
- **100% FLOSS**
  - FLOSS = Freely Licensed Open Source Software
- **Worldwide presence**

### **NewYork and Paris HQ**

Branch offices in  
California, Sweden,  
Japan and Australia



## Target Markets



## 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

### Customers

- Airbus
- Alenia
- Alstom Transport
- Ansaldo STS
- BAe
- Boeing
- EADS
- European Space Agency
- Eurocontrol
- JEOL
- Lockheed Martin
- MBDA
- Philips Semiconductor
- Raytheon
- Rockwell Collins
- SAAB
- General Electric
- Thales
- Thales Alenia Space

# GNAT Pro and AdaCore

A long history of serving avionics customers

Airbus  
Alenia  
BAE Systems  
Barco  
Boeing  
EADS  
Embraer  
Eurocopter  
Hamilton Sundstrand  
Harris  
Honeywell  
Kongsberg  
Lockheed Martin  
MBDA  
Rockwell Collins  
SAAB  
Sikorsky Aircraft  
Smiths Aerospace  
Thales

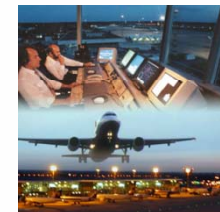
**Eurofighter Typhoon**  
Photo courtesy of BAE Systems

© 2007 Geoffrey Lee

**AdaCore**  
The GNAT Pro Company

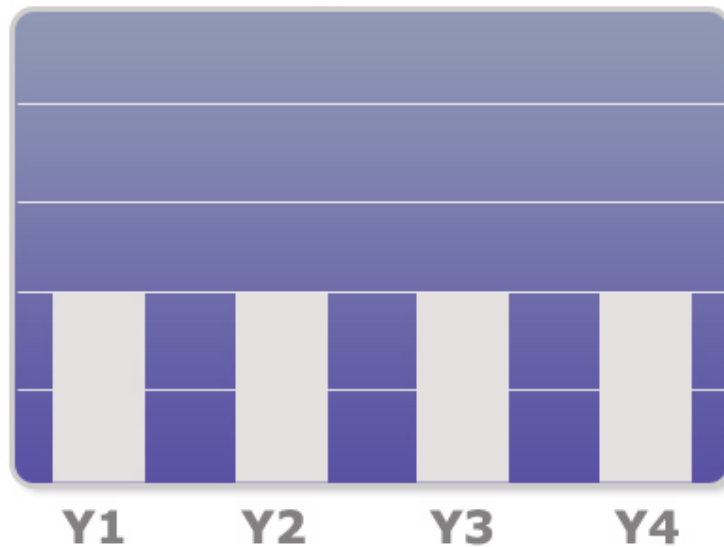
# GNAT Pro

Ada 2005 - Ada 95 - Ada 83

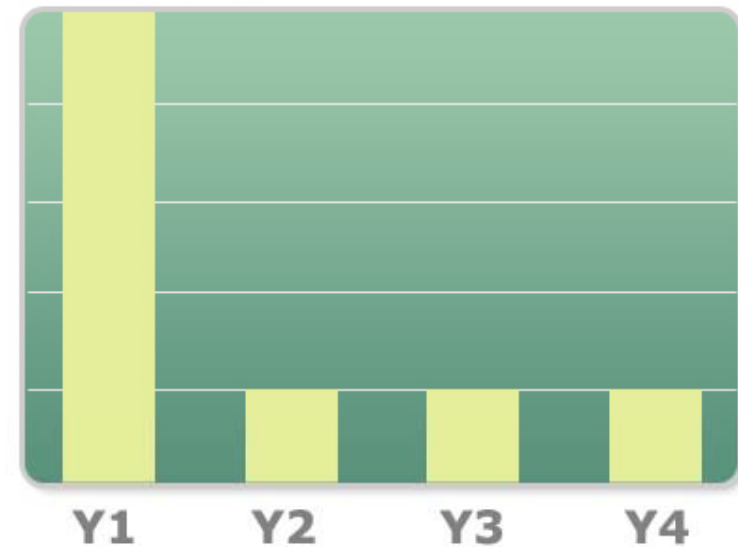


## Yearly Subscription Model

**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 Advice



## Tool Suite Support

---

- **Help with installation procedures**
- **Correction of problems encountered in the use of tools**
- **Workaround provided if a problem is encountered**
- **Pre-releases provided for high priority problems**

## Online Advice

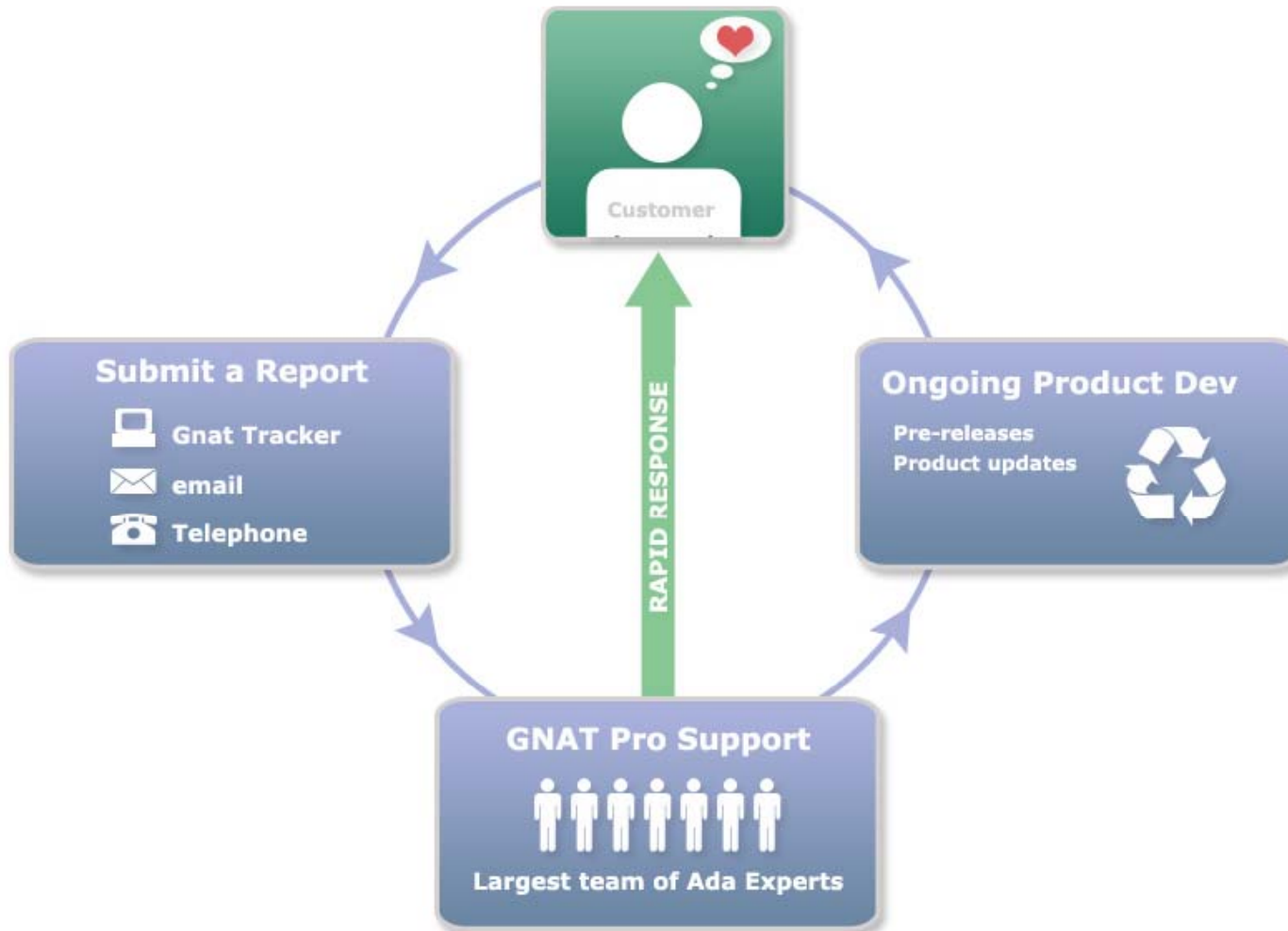
---

- Expert answers to questions about Ada 83 / 95 / 2005
- Suggestions for the effective use of Ada
- Suggestions on best use of GNAT tools for your project
- Help with all aspects of Ada software development

## Support + Online Advice: How it Works



## Support + Online Advice: How it Works



## Guaranteed Response Time

- 80% of requests are addressed in a few hours



# GNAT Tracker - Customer Web Server

The screenshot shows the GNAT Tracker customer web server interface. The header includes the AdaCore logo and navigation tabs for Dashboard, Tickets, Download, Wavefronts, Documentation, Developer, and Intern. The main content area is titled 'Dashboard' and shows account information for 'RAPITA SYSTEMS Ltd' with ID '1226'. A 'Subscription Summary' box indicates the subscription expires on Jan 20, 2008, for 5 Developers, with details for PowerPC ELF format. Below this are two tables: 'Open Tickets (Last 12 months)' and 'Recently Closed Tickets'. The 'Open Tickets' table lists four tickets with their descriptions, last messages, priorities, and waves. The 'Recently Closed Tickets' table lists three tickets with their descriptions, last messages, and waves. A right-hand sidebar contains a search bar, links to 'Create a New Ticket' and 'Download GNAT Pro', and a 'Useful Information' section with links to guides and policies. An 'AdaCore Developer Log' section at the bottom right lists recent releases and updates.

**GNAT Tracker**  
The GNAT PRO customer site

AdaCore  
The GNAT Pro Company

Log out ▾

Dashboard Tickets Download Wavefronts Documentation Developer Intern

## Dashboard

Account Information

RAPITA SYSTEMS Ltd | 1226

**Subscription Summary**  
Expires Jan 20, 2008: 5 Developers PowerPC ELF format (hosted on Windows), GNAT Pro High-Integrity Edition [Details ▾](#)

**Open Tickets** (Last 12 months)

Tickets	Description	Last Message	Priority	Wave
<a href="#">G201-006</a>	build scripts for g++	Mar 05, 2007	High	
<a href="#">G203-018</a>	multitasking in multiprocessor systems	Feb 28, 2007	Medium	
<a href="#">G118-023</a>	GPS: input of unicode characters in editor	Feb 24, 2007	Medium	
<a href="#">G110-009</a>	access to source code	Mar 03, 2007	Medium	

[All Tickets ▾](#)

**Recently Closed Tickets**

Tickets	Description	Last Message	Wave
<a href="#">G201-006</a>	running GNAT under Windows 2003	Feb 10, 2007	
<a href="#">G108-018</a>	support for non-Ada exceptions	Jan 21, 2007	
<a href="#">R22-023</a>	beginning with gtkada	Dec 28, 2006	

**Search**

[Create a New Ticket](#)

[Download GNAT Pro](#)

**Useful Information**

[Getting Started Guide](#)  
[GNAT Pro License](#)  
[GNAT Pro Support Policy](#)  
[AdaCore Company Site](#)

**AdaCore Developer Log**

Mar 02, 2007  
[GNAT Pro 6.0.1 Released](#)

Mar 02, 2007  
[No warnings for obsolescent Handling functions](#)

Mar 01, 2007  
[GNATmake checks for up-to-date .axe file](#)

## FLOSS and Business

---

- **Our success of the last 12 years**
- **Why did we succeed**
  - We helped people getting their job done
  - Our Business Model
  - The AdaCore virtuous circle
- **Repeating this success**

## **We succeeded because:**

---

- **We help Ada people get their job done**
- **We have the right business model for our market**
- **We are a great group of people**

## Getting the job done

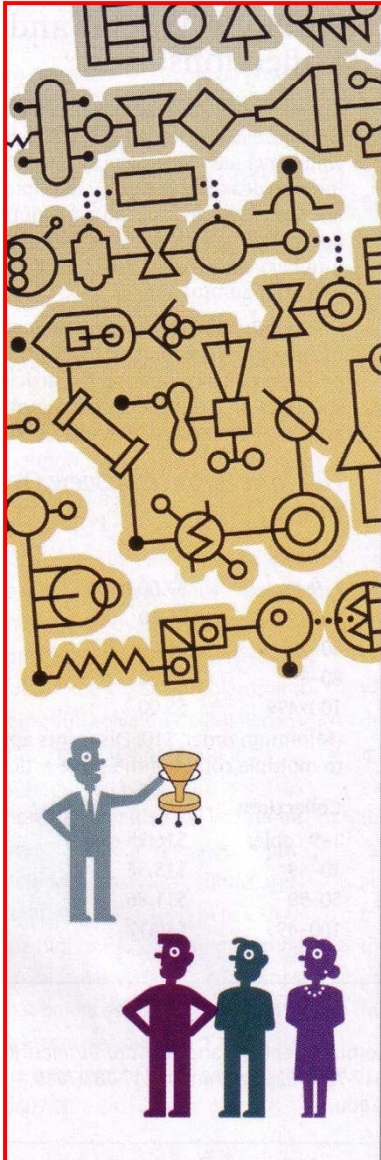
---

*“People don’t want a quarter-inch drill.”*

*“People want a quarter-inch hole.”*

Theodore Levitt

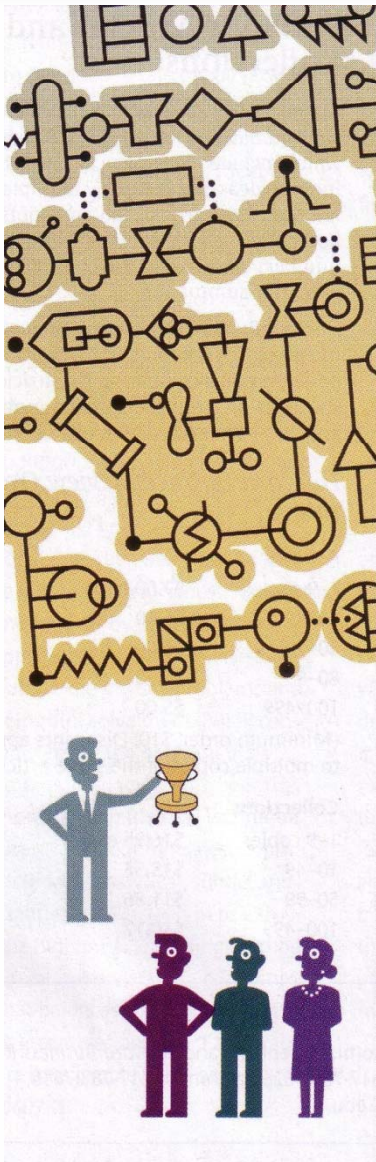
## Technology per se does not sell on the long run



## Marketing per se does not sell on the long run



## People buy products/services to get their job done



## What job do we help people with?

---

- Customer has software
  - Mostly written in Ada
  - That needs to be run on a computer
- 
- **Event requires change from old solution**
    - Port to new OS / hardware
    - Software update
    - Modernization of toolchain
    - Previous solution provider disappeared
    - ...

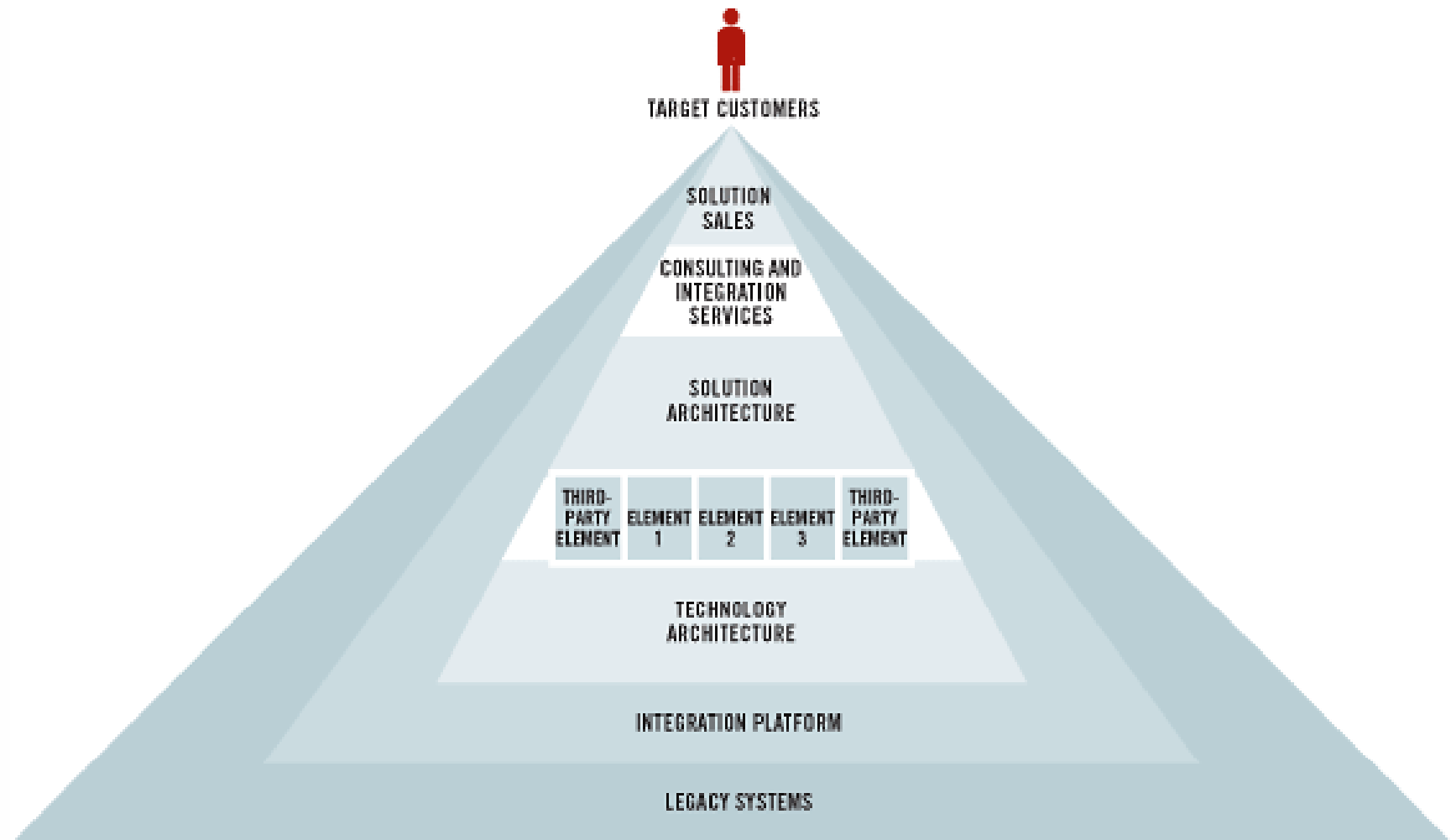
## The right business model for our market

---

- .... and for who we are !
- **Broadly speaking, 2 choices:**
  1. Complex Systems Model (300 customers x 30,000 = 9 M)
    - *McKinsey, Accenture, SUN, IBM*
  2. Volume Operations Model (30,000 customers x 300 = 9 M)
    - *Amazon, Google, Dell*

# The Complex-Systems Model

## COMPLEX-SYSTEMS MODEL

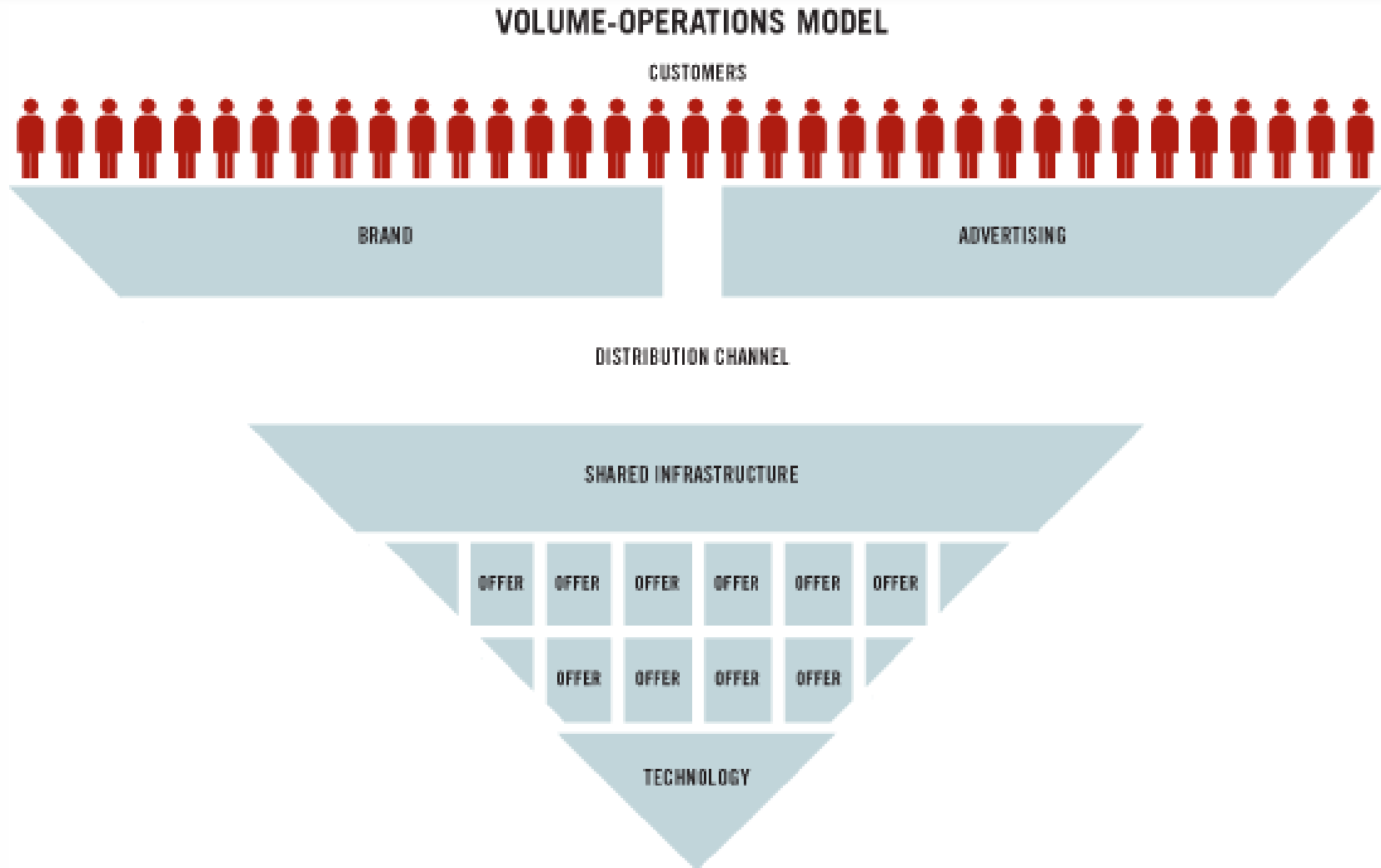


## The right business model for our market

---

- .... and for who we are !
- **Broadly speaking, 2 choices:**
  1. Complex Systems Model (300 customers x 30,000 = 9 M)  
*– McKinsey, Accenture, SUN, IBM*
  2. Volume Operations Model (30,000 customers x 300 = 9 M)  
*– Amazon, Google, Dell*

# Volume-Operations Model



## Repeating Our Success

---

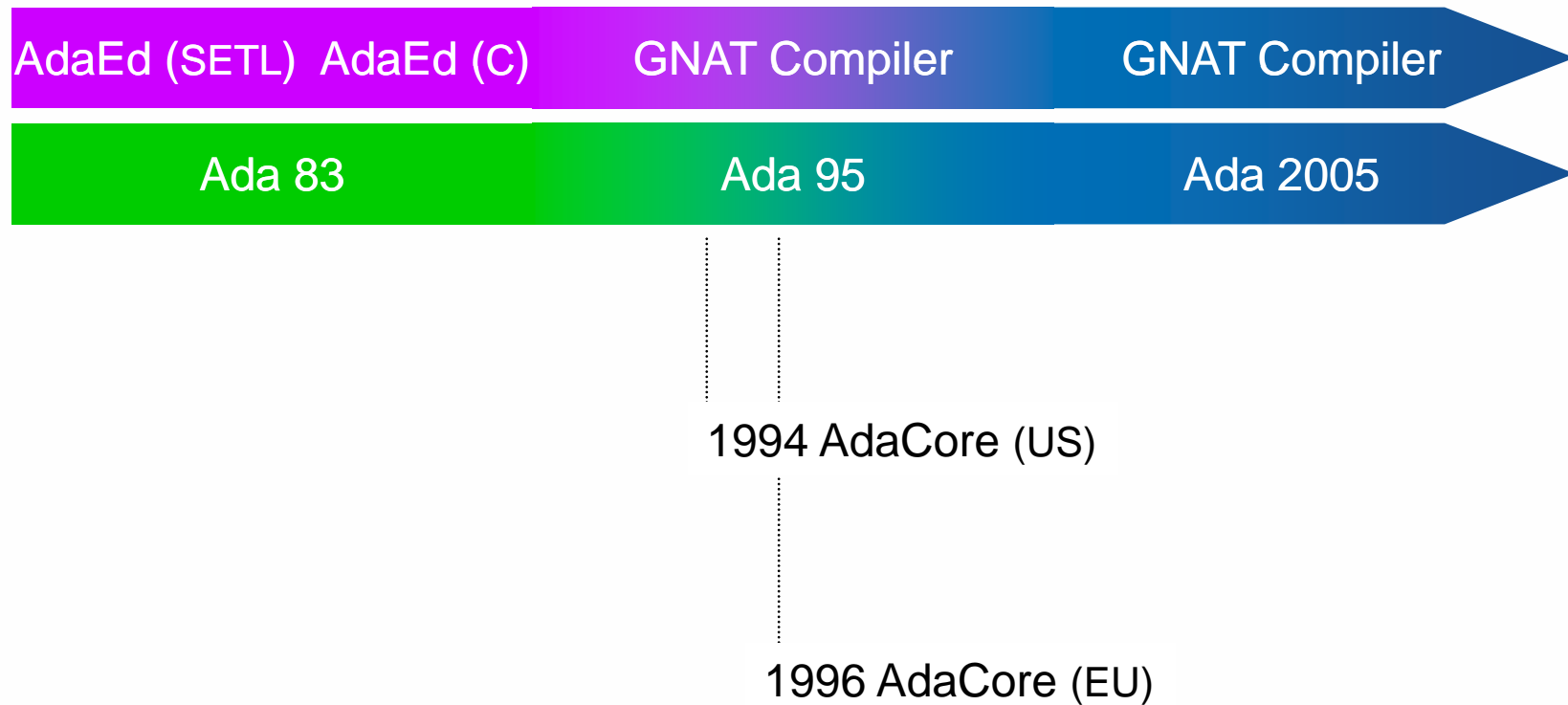
- **Help customers with other jobs**
  - **Stick to the current business model**
    - Few customers
    - High prices
- 
- **What we do should have a high value for customers**
  - **Focus on our collective strengths**

## Helping customers with other jobs



- How did we happen to help them in the 1<sup>st</sup> place?

## Where do we come from?



## The initial group



**Tech group**

**With specific interest:**

**Develop and run Ada code**

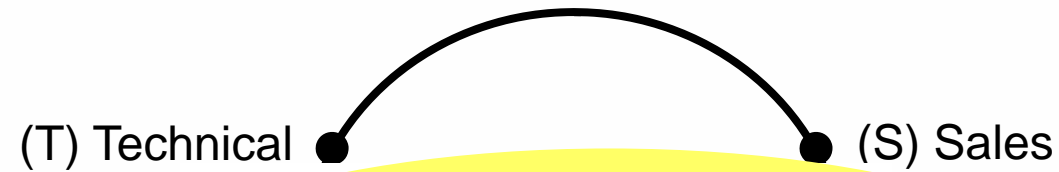
**+ It'd be nice to sell what we do  
(Ed's "Good Woman of Setzuan")**

## How did we happen to help customers in the 1<sup>st</sup> place?

---

- **Serendipity**
  - **We helped ourselves**
  - **We solved problems we were interested in**
  - **This happened to solve other people's problems**
-

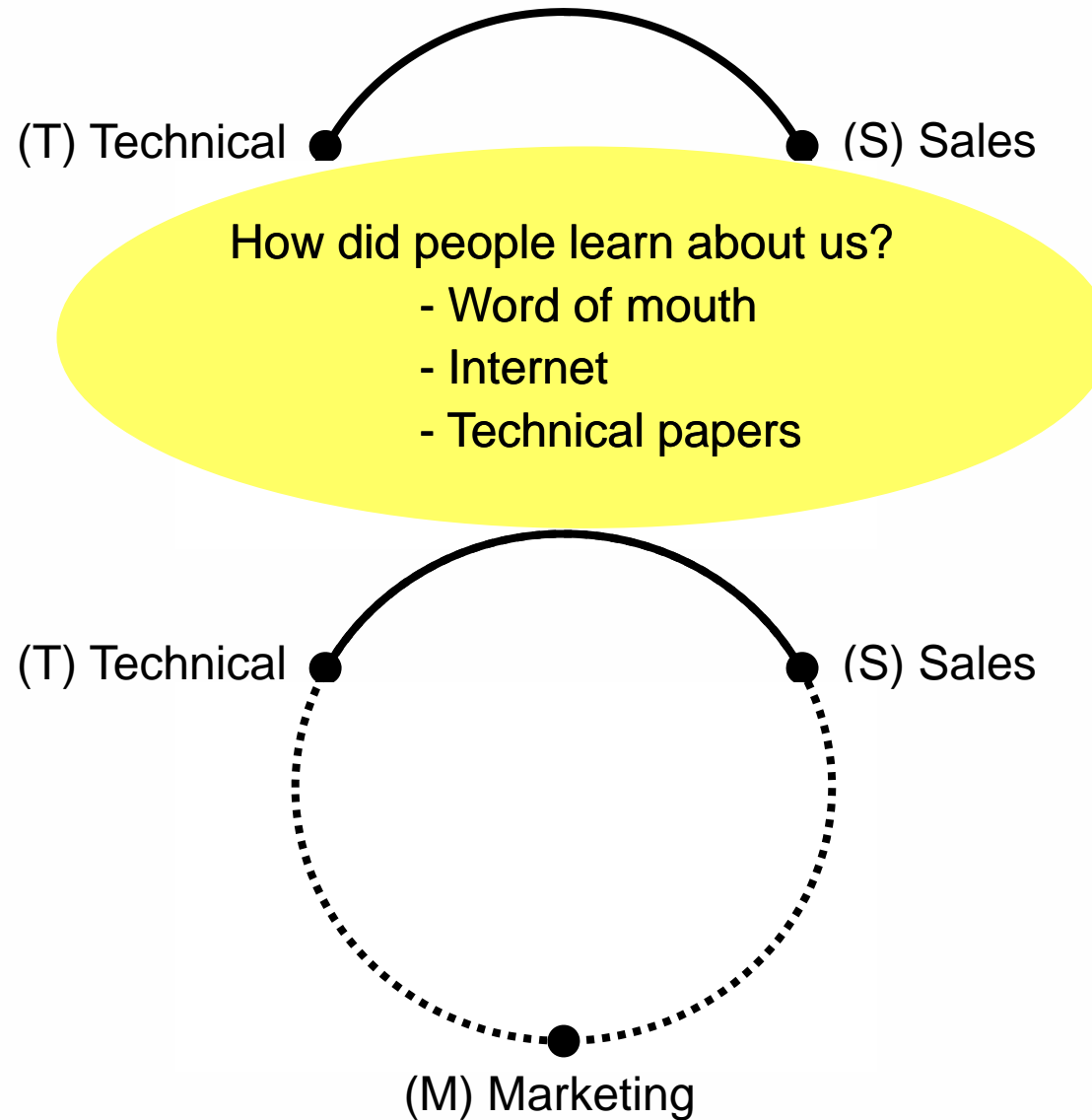
## AdaCore: The Beginning



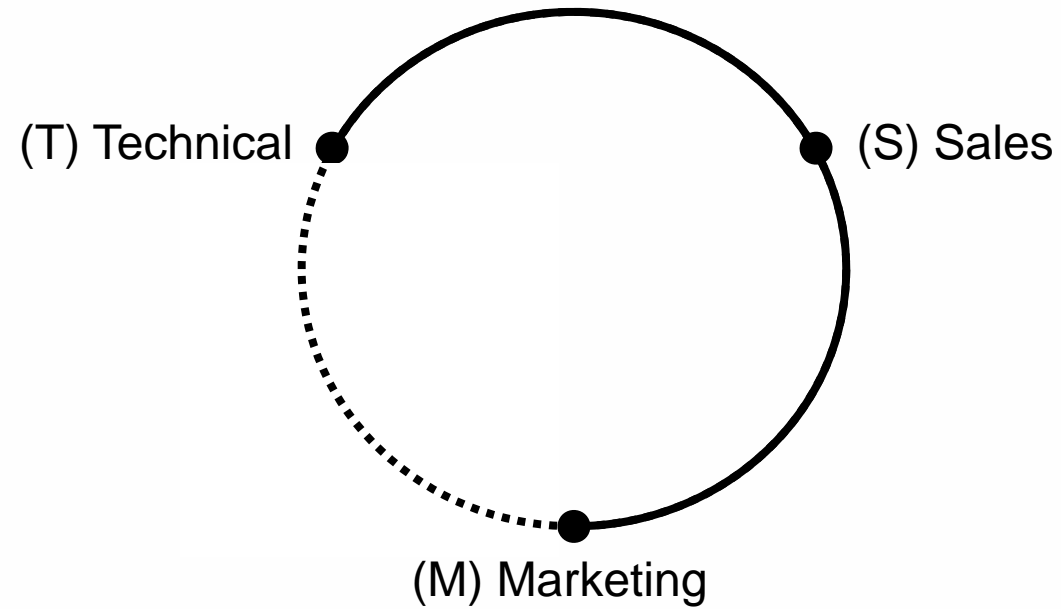
How did people learn about us?

- Word of mouth
- Internet
- Technical papers

## AdaCore: The Beginning

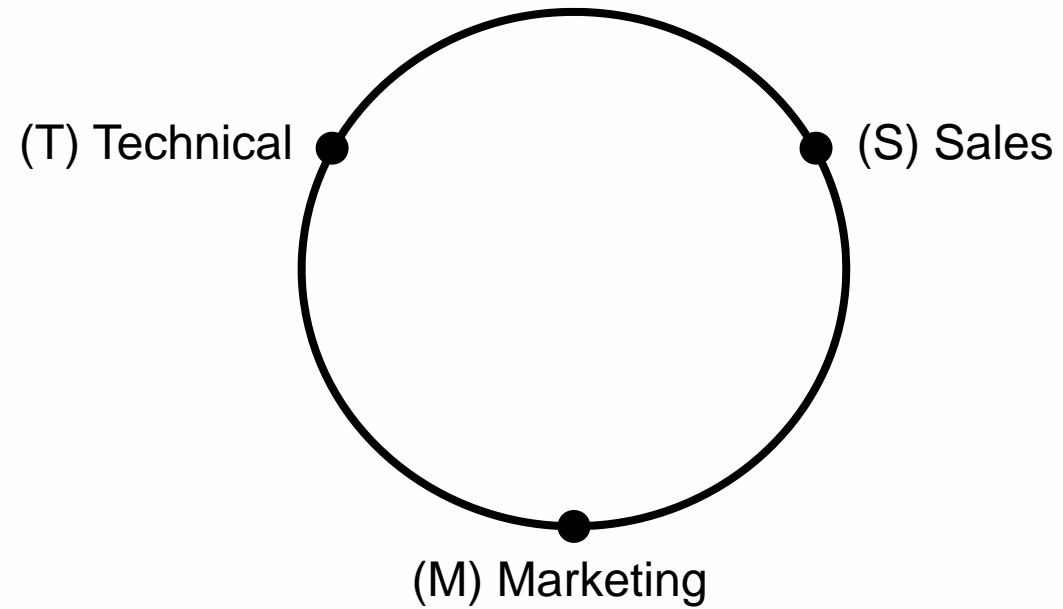


## AdaCore ca 2000



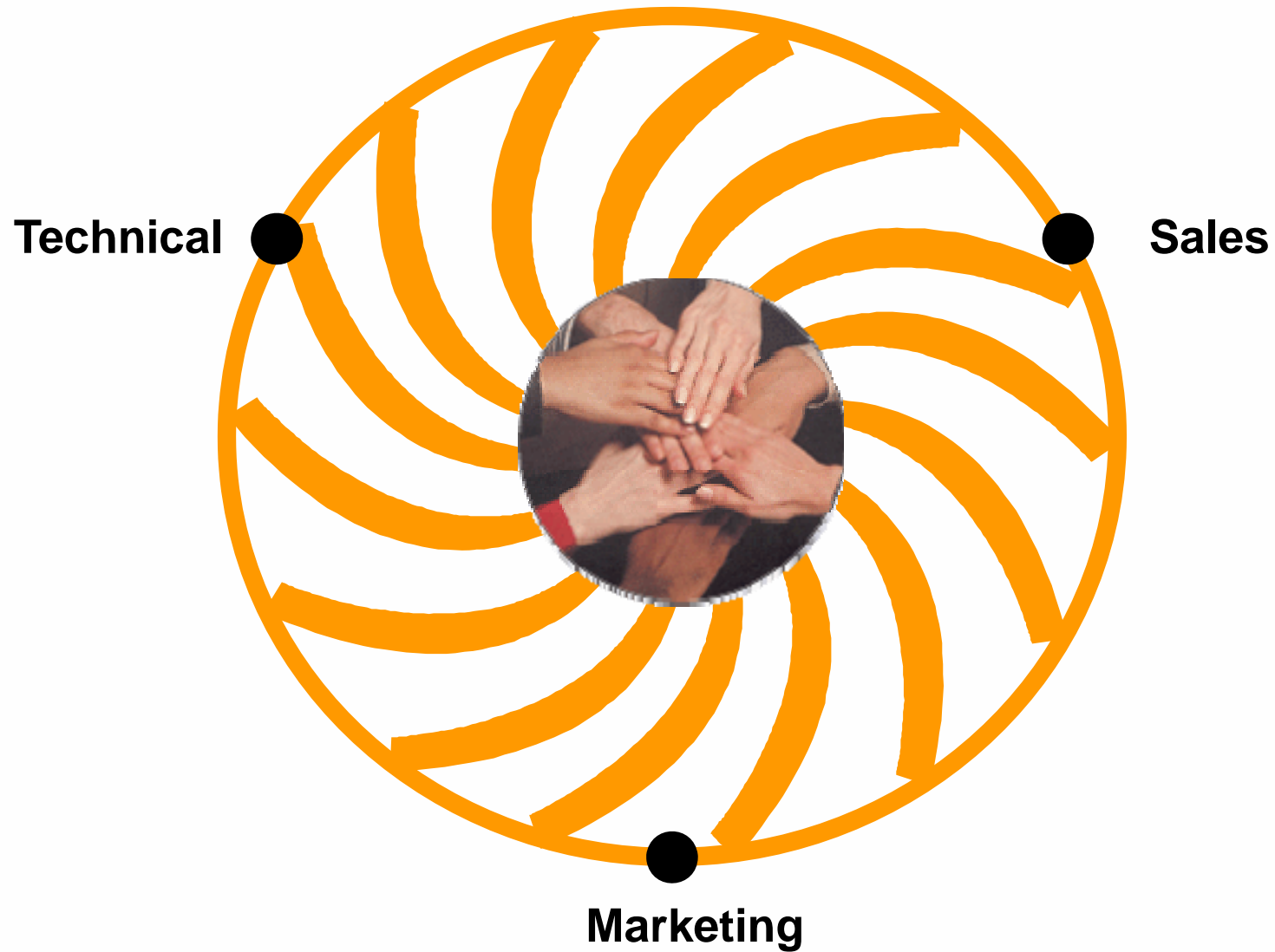
## AdaCore 2006

---



- **As we explore the realms of JOBs to solve**
- **The AdaCore circle of life is key**
- **In finding the “right” problem to solve**

## AdaCore Circle of Life



## Conclusion: Repeating Our Success

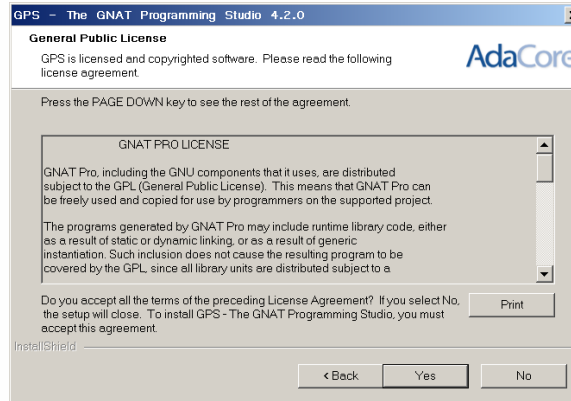
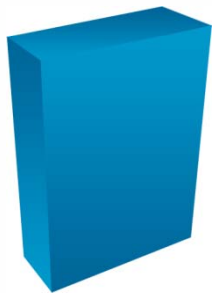
---

- **Help customers with other jobs**
  - Finding which - is the hard part 😊
- **Stick to the current business model**
  - Few customers
  - High prices
- **Leverage on the AdaCore circle of life**

# What I Tell Industrial Users

---

# FLOSS



## FLOSS License

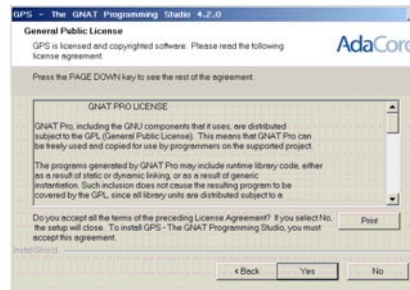
- Free to run
- Free to look at sources
- Free to change



## FLOSS is Freedom



# Isn't FLOSS Free-of-Charge ???



**FLOSS License**





## I can ask the “Community”



## Who is the “Community” ?



## The GCC Example



## “Communities” are really **Coops**



- **Industrial partners**
- **Cooperate on **technology****
  - Not products
- **Cost reduction**
  - By sharing resources
  - To solve a common problem
- **Peter Kropotkin (1902)**
  - Mutual aid: A factor of evolution

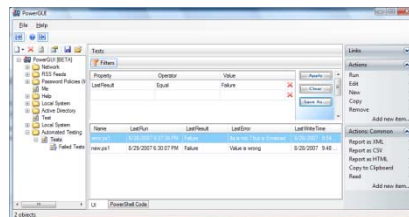
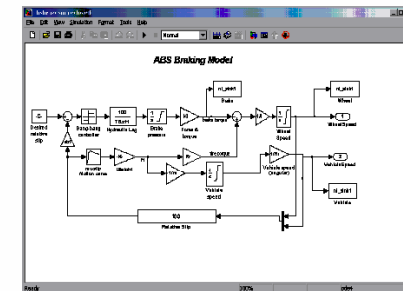
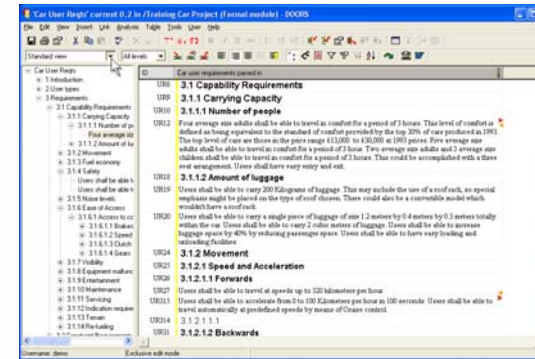
## Industrial ERTS



- **Complex**
- **Long life-span**
- **Harsh operating conditions**
- **Safety issues** (DO-178B, IEC 61508, 00-55/56, EN 50128, ...)

# Building Industrial ERTS

- Configuration management
- Requirements tracking
- Modeling tools
- Software development tools
- Testing tools
- Build infrastructure
- ...



## FLOSS for ERTS draws from **many Coops**

- Configuration management
- Requirements tracking
- Modeling tools
- Software development tools
- Testing tools
- Build infrastructure
- ...



## FLOSS Gives You Many Options

---

### Getting the FLOSS

- Download the FLOSS as-is
- Buy the FLOSS from a vendor

For each FLOSS  
decide what to do

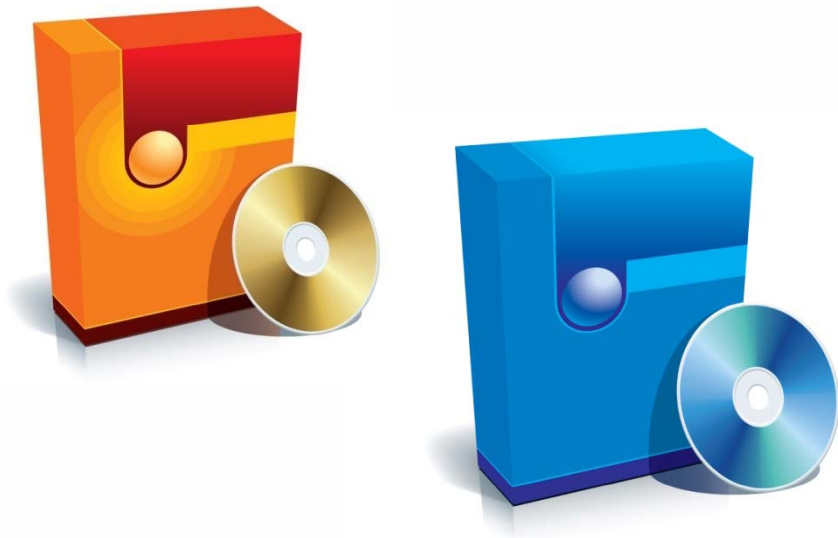
### Getting help

- Do nothing
- Build in-house expertise
- Become part of the coop
- Purchasing Services

# The Job of a FLOSS VENDOR



## The Job of a FLOSS VENDOR



## Purchasing FLOSS and Cost Reduction



- **Industrial cooperation**
- **Cost reduction**
- **Can be passed on**
  - Cheaper products
  - More value for the same price
  - More services for the same price
- **Cheaper or you get more**
  - or both



## Your Choices: Summary

---

- Download the FLOSS and use it
- Build in-house expertise
- Become part of the coop
- Purchase

**COST** vs. **RISK** analysis

## The Virtuous Circle of FLOSS & Business



- **Interests aligned**
  - Customers
  - Employees
  - Long-term investors
- **Long-term value-creation**

## Research Topics

---

- **Going beyond the sources**
  - Software is more than the sources
  - Extending the concept of OSS for the coming decade
- **Creating FLOSS business model for an ecosystem**

## To Share the Adventure

---

- **For an internship**
  - [hainque@adacore.com](mailto:hainque@adacore.com)
  - [gasperoni@adacore.com](mailto:gasperoni@adacore.com)