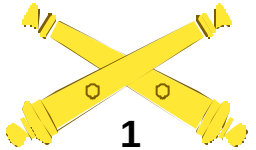
Ada: The Right Choice for Reliable Software, Tri-Ada '97

John A. Hamilton, Jr., Ph.D.
Lieutenant Colonel, United States Army
Research Director

D/EECS
US Military Academy
West Point, NY 10996

Fax: (914) 938-5956 (DSN 688)
dj7560@eecs1.eecs.usma.edu
hamilton@exmail.usma.edu

Programming Languages Do Make a Difference



Summary of NRC recommendations

- **Require Ada for DOD warfighting software.**
- **Drop Ada requirement for other DOD software.**
- **Invest $15M/year for Ada infrastructure - or drop Ada requirement entirely.**
- **Program language selection should be part of a rational software engineering process.**

Warfighting Software: An Unofficial View

- Clearly, this is the software that the Defense Community is most concerned with.
- Warfighting software is not COTS.
- In my view, any system that can affect battlefield performance is a warfighting system.
- It is more than just embedded systems. Information systems (such as AFATDS) will interact between both embedded systems and warfighters.

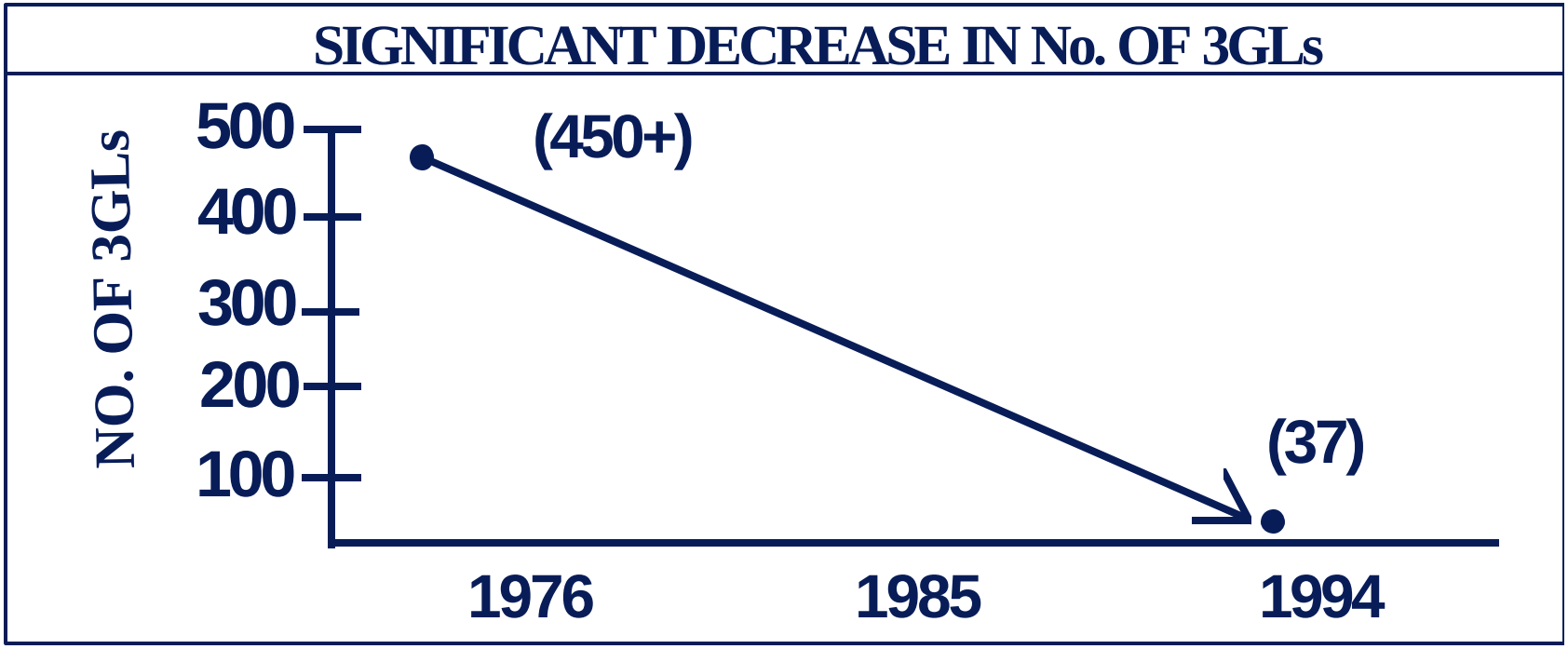
Non-warfighting software

- A BOQ reservation system that is unreliable is inconvenient, but not a war stopper.
- Military requirements for non-warfighting custom software should be minimal.
- Just because a system operates strictly in a CONUS garrison environment does not mean it is not a warfighting system.
- Example: a personnel mobilization system that can erroneously list a reservist as being hospitalized for minor surgery for three continuous years denies a warfighting asset to a theater of operations.

A viable Ada infrastructure is a military necessity

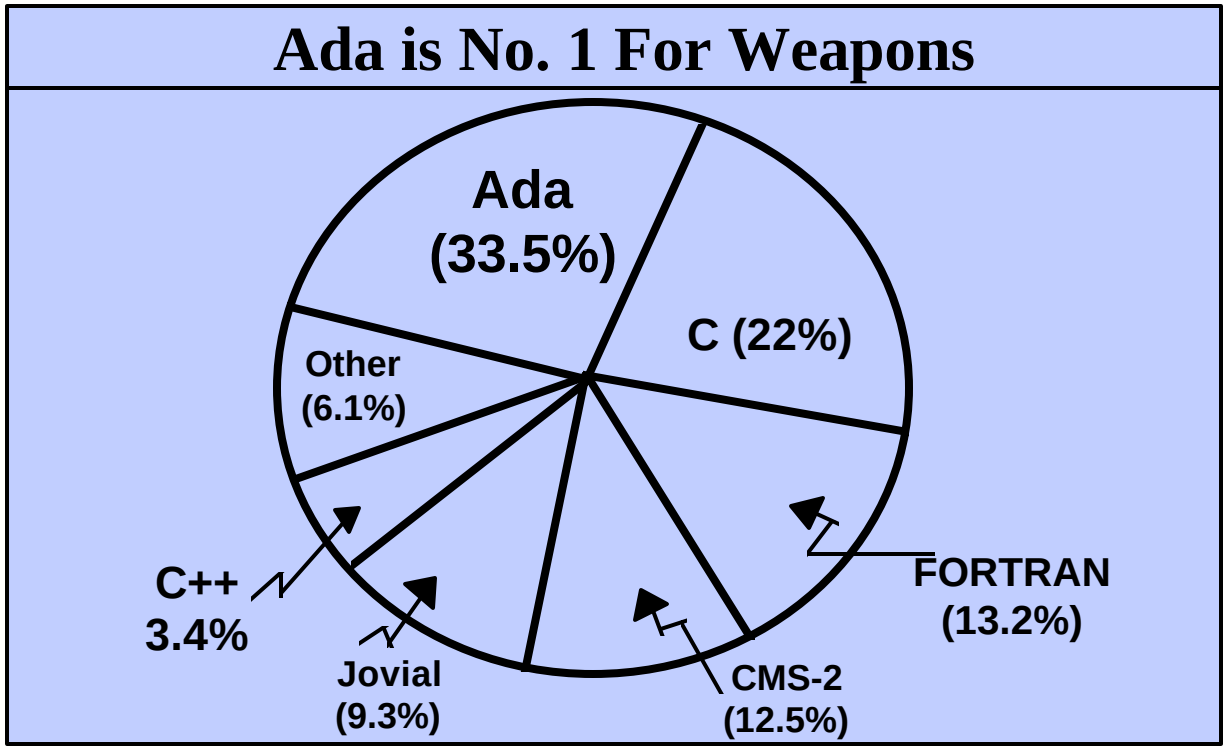
- **“Fifty million lines of Ada warfighting code will become a liability without a robust Ada infrastructure.”**
- **What happens when artillery fire control systems cannot be modified because the software is not maintainable?**
- **What happens when critical systems such as Field Artillery survey computers have to be updated due to unexpectedly extreme climactic conditions?**
- **Inability to quickly and adequately maintain combat systems is a potential war stopper.**

The Number of Programming Languages used in DOD Declines



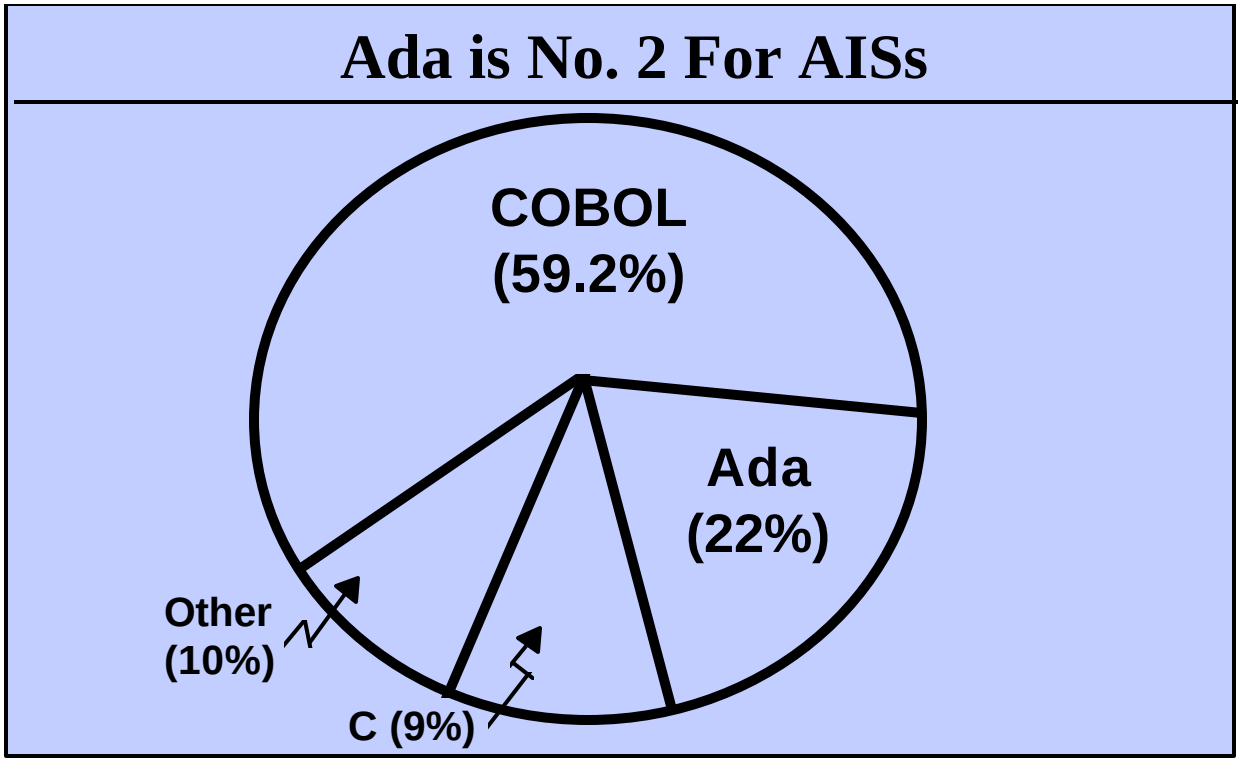
92% reduction in different programming languages in 20 years

Programming Language Use in DOD Today: Weapon Systems



Breakout of programming language usage in DOD weapons systems.

Programming Language Use in DOD: Automated Information Sys.



Programming language usage in DOD automated information systems.

Superior technical capabilities

- "In warfighting applications, Ada's technical capabilities for building real-time, high assurance custom software are generally superior to those of other programming languages."
- Criteria used by NRC:
  - High-assurance criteria
    - Enforcement of modularity
    - Support for user-defined abstraction
    - Management of pointers
    - Management of software faults
  - Real-time criteria
    - Safe static data allocation
    - Predictability of meeting deadlines
    - Interaction among threads of control

Reliability Counts

- A one degree error at a range of 40 kilometers equals a 700 meter lateral deviation.
- The precision engagement imperative of Joint Vision 2010 in particular requires high reliability.

Lifecycle Considerations

- **Military software systems continue to have long lifecycles.**
- **Software maintenance is still the greatest software cost over the software lifecycle.**
- **Ada virtually always wins cost comparisons when maintenance is considered.**

Ada as a teaching language

- 47 percent increase in institutions offering Ada courses in past 3 years.
- Ada seen as a viable replacement for Pascal.
- Educational literature report severe difficulties with academic use of lower-level languages.
- Excellent Ada resources available in the public domain.

Past & Present Contexts for Ada in the DOD

Past

- DOD dominant software player
- Secondary role in DOD for software
- No existing code written in Ada
- DOD committed to major Ada development investment

Present

- DOD large software player
- Software plays primary role: key to DOD goal of information dominance
- 50 million lines of DOD weapons systems written in Ada
- DOD preparing to drop its investment in sustaining Ada

DOD Software Domains

Warfighting Software	*Commercially Dominated*
• Weapon control, electronic warfare, real-time sensor processing, battlefield-unique communications	• Office and management support, routine operations, asset status monitoring, logistics, medicine, backbone communications
• Domain expertise mostly within DOD community	• Domain expertise mostly commercial
• Mostly custom software	• Mostly COTS-driven
• Software in Ada achieved critical mass	• Very little software in Ada

Software Maintenance

- DOD cost estimates for maintenance over the software lifecycle range from 67% to more than 90%.
- Like automobiles, long term utilization increases the overall return on investment.
- Fewer new weapons starts means we will upgrade and modernize the systems we have fielded.
- We can verify the existence of fifty million lines of Ada code in critical warfighting systems.

Limitations on Commercial Software

- Not available for many domains.
- Sold as is with no warranty and no independent code verification.
- Source code often not available or only available at significant cost.
- Modification of a COTS component by DOD means that it is no longer off-the-shelf and may be incompatible with a vendor's future releases.

COTS has Limitations

- COTS applications are often brittle, proprietary and incomplete.
- We cannot buy weapons systems off the shelf.
- Modifying commercial applications through the use of custom code is often the worst of both worlds.
- We will not win wars through superior word processing.

Final Observations on Trends

- Common commercial programming languages will evolve to meet military requirements.
- Software maintenance requirements will dictate the use of public standard languages.
- 3GL-style programming languages will look more and more like Ada.
- CASE/4GLs will evolve to general-purpose usefulness, but this will take longer than people expect.

Why the DOD is interested in Programming Languages

- Commercial programming languages do not always meet military requirements.
- There exist critical warfighting systems written in Ada that must continue to be supported.
- Ada will be playing a key role in the Defense Department well into the 21st century regardless of what happens in 1997.

Ada 95 Today

- Ada usage in the DOD is impressive, the M1A2 tank, the Aegis system, the F-22 are Ada systems.
- Ada is alive and well in our warfighting systems.
- For the Defense Department this essentially means that the Ada debate is moot.
- Ada will be playing a key role in the Defense Department well into the 21st century regardless of what happens in 1997.

Ada 95 vs. C++

- This is the wrong question on a variety of levels.
- First, the clear trend in programming languages is towards higher levels of abstraction.
- This trend really works against C and that is one reason why the use of C is declining.
- Higher levels of abstraction supported in C++ are notoriously non-standard. A very interesting illustration of this problem appears in the May, 1997 issue of CrossTalk

Reliability is Important

- **Commercial software standards are NOT good enough.**
- **A 700 meter range error can easily kill US/Allied soldiers.**
- **Software that works 99% of the time built using “commercial best practices” will not impress a Gold Star Mother.**

Compiler Validation

- Standards produce interoperability and lower costs.
- Formal validation answers the question of how well a compiler conforms to a standard.
- Ada language features reduce errors and provide for high reliability.
- Validation provides high assurance that the reliable language features are implemented.

The DOD Environment

- It is widely believed (incorrectly) that the United States no longer faces significant military threats.
- Requirements will continue to outstrip resources for the foreseeable future.
- Y2K challenges may well absorb most of the limited resources available.
- Federal budget pressures will continue to force short term decisionmaking because uncertainties in the out years continue to increase.

DOD Software Trends

- DOD requirements for software are greater than available resources and those requirements are increasing.
- DOD software will continue to have long lifecycles.
- Software reliability requirements are increasing.
- Commercial, Off-The-Shelf, (COTS) software solutions sought where possible.

The Future of Ada in the DOD

- The need for DOD software standards, including programming language standards, has not diminished.
- Despite advances in COTS and 4GLs, there are many military requirements that cannot be satisfied with COTS.
- DOD Program Managers need education, training and information provided regarding Ada capabilities and resources, in other words an Ada Joint Program Office.

Competitive Edge

- It is difficult to put a price on reliability.
- Validation is not only an important tool to protect the government's interest, it can be a useful marketing tool as well.
- Non-proprietary reuse and government-responsible software maintenance are not design parameters for COTS.

CONCLUSIONS

- As noted in the NRC Report, in military applications, Ada95 is often the best solution when reliability is considered.
- In a resource-constrained environment, the best technology does not necessarily win.
- Successful Ada initiatives will have to show program savings up front.
- Failure to maintain an adequate Ada industrial infrastructure may result in the inability to sustain critical warfighting systems.