

Programming Language Policy in the DoD: After the Ada Mandate



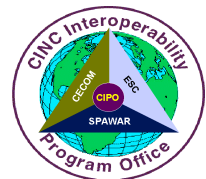
John A. (Drew) Hamilton, Jr., Ph.D.
Lieutenant Colonel, Field Artillery
Deputy Director, SPAWAR
CinC Interoperability Program Office

Code 05J
4301 Pacific Highway
San Diego, CA 92110-3127
(619) 524-7225 voice
(619) 524-7224 fax



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



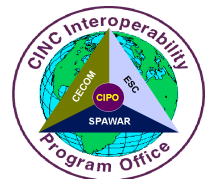
The Here and Now

- We are unlikely to return to the programming language jungle of the 1970s without aggressive mismanagement on the part of DOD.
- Proprietary implementations of compilers can still cause major headaches later.
- “Later” is not on the scopes of many DOD IT organizations.
- Just because you **cannot** buy COTS for many military-specific applications does not mean that COTS is still not the “preferred solution.”



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



DOD Programming Language Objectives

- **Reliability**

Software used to kill people and break things **MUST** be reliable.

- **Maintainability**

Despite the rapid rate of change in commercial software applications, custom-coded, DOD warfighting software will continue to have long lifecycles.

Increasingly, weapon upgrades are software based.

- **Reuse**

Reuse at the function/procedure library level is the realistic target.

- **Portability**

The DOD will remain a heterogeneous computing environment.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



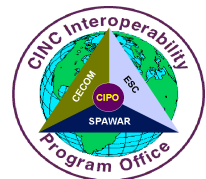
Reliability

- Research in the academic community demonstrates that programming language characteristics can significantly limit the nature and scope of errors.
- Significant risk avoidance can be achieved by using commercially available compilers for which there is an established track record for errors common to a particular language.
 - ex. **overwriting main memory addresses.**
- Standardized compilers, those whose compliance to a standard has actually been validated by testing against a standard suite, pose the least risk in terms of reliability.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



Maintainability

- Programming language characteristics, programming style and the problem domain all impact on the future maintainability of the code.
- Compilers which adhere to a standard and which prohibit subsets or supersets of the standard, reduce the learning curve for maintenance programmers.
- Language readability greatly enhances maintainability.
- Strongly typed languages are easier to trace and therefore to maintain.
- Published style guides facilitate understanding the source code by other than the original code developers.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



Reuse

- Reuse at the function/procedure library level is the realistic target.
- Commonly used libraries decrease risk since errors may have surfaced in earlier employment of the library function.
- Libraries which only have publicly available APIs are riskier since the complete functionality of a library package may not be verifiable without a complete review of the source code.
- Major non-technical issues associated with reuse.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



Portability

- DOD will remain a heterogeneous computing environment.
- Compilers that do not adhere to any recognized standard such as those of the IEEE or the ISO are often tied to a single hardware/OS platform.
- Additionally, commercial compilers often include powerful, but non-standard extensions.

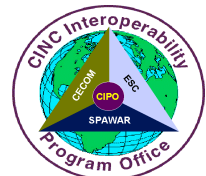
Early estimates of portability may be achieved by recompiling the source code using another vendor's compiler targeted for the same OS/hardware platform as the first compiler. This will largely identify the use of non-standard constructs.

Portability between operating systems can be measured early by recompiling the source code with a compiler targeted for another operating system. This will identify the use of OS-specific system calls.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99



Conclusions

- We are unlikely to see a return of a programming language mandate.
- Programming language selection should be an *engineering* decision.
- When engineering criteria such as reliability, maintainability, reuse and portability are considered, Ada 95 will often emerge as a clear winner.
- Do not underestimate the influence of trade journals and pundits on DoD policy.
- *Publicized* success of Ada in non-government related activities most likely to influence DoD decisionmakers.



SPAWAR CinC Interoperability Program Office

www.drew-hamilton.com/stc99

