

A COTS-Based Replacement Strategy for Aging Avionics Computers

Jahn A. Luke

Air Force Research Laboratory/Information Directorate

Douglas G. Haldeman and William J. Cannon

TRW Avionics System Division/Avionics Engineering Centers

This article describes a commercial off-the-shelf (COTS)-based form, fit, function, and interface replacement strategy for legacy avionics computers and embedded information systems that can reuse existing software code as is while providing a flexible framework for incremental upgrades and managed change. It is based on a real-time embedded software technology that executes legacy binary code on the latest generation COTS microprocessors. This scaleable technology, developed by TRW and sponsored in part by the Air Force Research Laboratory, demonstrates performance improvements of five to 20 times that of the legacy avionics computer that it replaces. It also promises a four-fold decrease in cost and schedule over rewriting the code, and provides a known, good starting point for incremental upgrades of the embedded flight software. Code revalidation cost and risk are minimized since the structure of the embedded code is not changed, allowing the replacement computer to be retested at the black-box level using existing qualification tests.

Military aircraft are active years longer than originally anticipated. The avionics computers on these aging aircraft are expensive to maintain due to parts obsolescence, and require additional processing and memory to handle changing requirements. As a result, old computer hardware needs to be replaced with newer, more capable microprocessor technology. However, incompatibility issues require that embedded software be rewritten. It is estimated that billions of upgrade-dollars could be saved if new computers could execute the old embedded code along with any new code to be added.

TRW has developed a generic commercial off-the-shelf (COTS)-based software technology called Reconfigurable Processor for Legacy Avionics Code Execution (RePLACE) that capitalizes on technology advances to eliminate costs associated with rewriting legacy software to execute on new hardware. This software allows a user to 1) replace obsolete computers with commercial processor-based hardware, 2) continue to use existing

operational flight plan (OFP) without modification, and 3) incrementally upgrade hardware and software to support new and modified capabilities. RePLACE COTS software allows Department of Defense (DoD) dollars to be spent on solving supportability and performance problems and adding new needed capabilities, rather than recapturing current capabilities in new hardware.

Figure 1 depicts using the COTS software to migrate from a current legacy avionics line-replaceable unit (LRU) to a new COTS-based replacement box (CRB). Typically, today's legacy avionics and embedded information systems are based on custom hardware and proprietary back-planes with an obsolete 16-bit instruction set. Little or no "modern" higher order language (HOL) support is available to support software for this equipment. In addition, these legacy systems are often limited in terms of throughput and memory.

A COTS-based, open-systems replacement strategy provides a low cost of entry strategy for COTS microprocessor tech-

nology insertion. Both the legacy instruction set architecture (ISA) as well as the new native ISA can be executed. Native code execution is supported by advanced HOLs such as Ada95 and C++. In addition, legacy code can execute much faster in the new hardware and utilize more memory than is available for needed upgrades.

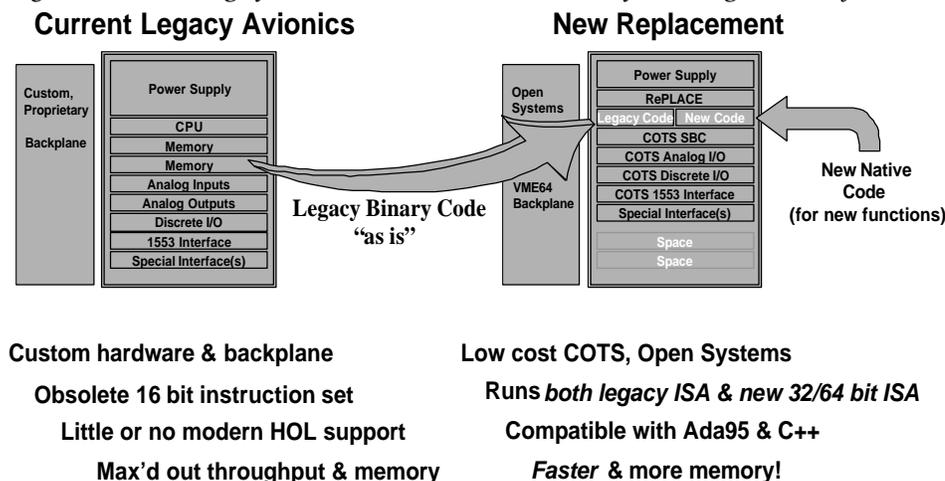
A software perspective of RePLACE is illustrated in Figure 2. Sitting on top of the COTS microprocessor and Portable Operating System Interface (POSIX)-compliant real-time operation system (RTOS) are two virtual machines: the legacy virtual machine and the native virtual machine. Within the legacy virtual machine space are four key software components: the legacy instruction set engine, the legacy OFP code binary image, the input/output (I/O) mapping software, and the virtual component environment (VCE).

The legacy instruction set engine contains unique cache-optimized code developed by TRW to execute the legacy binary OFP code. The I/O mapping software maps the new COTS interface devices to the legacy interface devices. The VCE allows for the efficient transition between the native and legacy virtual environments allowing the transfer of data and concurrent execution of both legacy and new native code.

The key features may be summarized as follows:

- Relies on using state-of-the-art COTS microprocessors and open system standards that improve both the legacy system's producibility and supportability.
- Runs legacy code from five to 20 times faster than the original legacy system by using a unique cache-optimized approach. This provides extra comput-

Figure 1: Current Legacy Avionics to a New COTS-Based by Utilizing COTS Software



ing power for new functions. In addition, the performance of the legacy virtual machine is linearly scaleable with the performance of the COTS microprocessor technology that is being used. That is, as the internal clock speed of the microprocessor chip goes up, the legacy instruction/execution speed increases linearly.

- Makes the instruction set engine adaptable to diverse instruction set architectures, including, for example, MIL-STD-1750A, Z-8002, and AN/AYK-14A. This makes possible the replacement of multiple diverse legacy avionics LRUs with a common set of avionics hardware modules on a given platform or across different platforms.
- Provides I/O mapping software that exactly matches the new I/O COTS devices to the legacy I/O devices, providing a *drop-in* environment for the unmodified legacy OFP with little or no knowledge required of the original code.
- Executes both legacy and new native software concurrently in separate virtual spaces. This promotes incremental addition of new capabilities and gradual transition to new code.
- Provides instruction execution speed matching that can be initiated in critical sections of the OFP to provide legacy OFP timing adjustments for timing sensitive code.
- Allows practical real-time non-intrusive (RTNI) monitoring of legacy software built into the replacement avionics, dramatically enhancing the observability and testability of the embedded legacy software.

In order to illustrate the concurrent execution of legacy and native code in a RePLACE dual instruction set computer (DISC) environment, Figure 3 depicts the legacy code remaining in control while new software enhancements are introduced. On the far left are two time-lines showing the various rate groups processing tasks running in the legacy machine and in the CRB.

In the case of the CRB, the original rate groups are executed in a much shorter time frame within any given minor cycle. This leaves additional processor throughput at the end of each minor cycle to add new software running in the native ISA. Through the VCE context switch mechanism, referred to as a *think*, new native code can be introduced to replace existing legacy code or added to the existing legacy code.

Alternately, event flags can be set to augment the legacy code thread as illus-

trated. Note that legacy instruction execution speed matching can also be introduced for timing-sensitive code. In addition, the technology includes the capability to disable legacy code outputs without legacy OFP cognizance, providing a convenient mechanism to switch off functions in the legacy code without being intimately familiar with the legacy code structure. In all cases, the original legacy binary OFP code remains intact with *no* changes.

Another key component of the COTS software strategy is the availability of a source level, symbolic user console for system developers. A tool has been developed to facilitate the use of RePLACE in modern microprocessor technologies and is referred to as the virtual integration environment workstation, or VIEWstation. It is tightly integrated with the COTS native code integrated development environment and commercial tools that are selected to support new native code development.

VIEWstation is loosely integrated with the legacy code tools. It provides a source level, symbolic configurator tool to assist the software developers in mixing and matching legacy and native code; it also provides in-target debugging and RTNI monitoring of the legacy code. VIEWstation incorporates COTS graphical data analysis tools and an interactive symbol browser/editor. Examples of VIEWstation use include downloading and disassembling software binaries, displaying real-time debug and monitoring data, performing and displaying timing

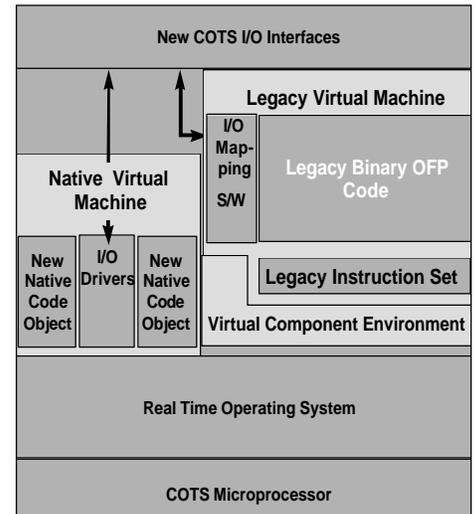


Figure 2: A Software Perspective of RePlace

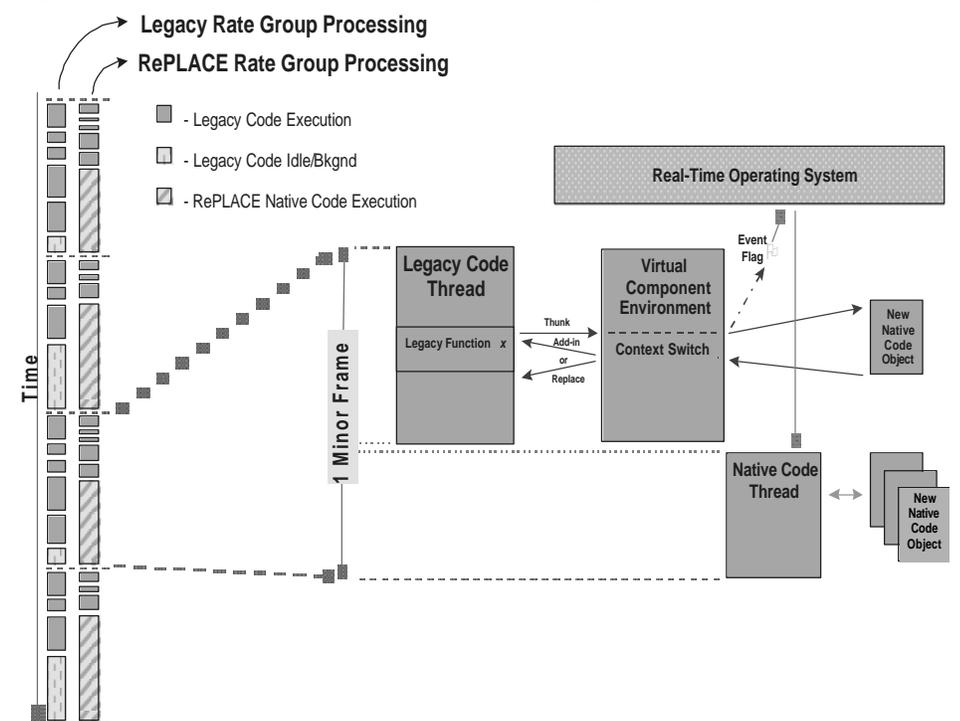
characteristics, creating and deleting events, and developing *add-in* code that is executed in response to specific user-specified events in the legacy code.

Results

Demonstrations of RePLACE executing legacy binary OFPs have been conducted in conjunction with AFRL for the following DoD aircraft subsystems:

- F-16 Heads-Up Display.
- F-16 Advanced Central Interface Unit (stores management processor).
- F-16 General Avionics Computer (fire control computer).
- AC-130H Gunship Mission Computer (SKC-3007A).
- C-17 Communication Control Unit.

Figure 3: New Software Enhancements Introduced to Legacy Code



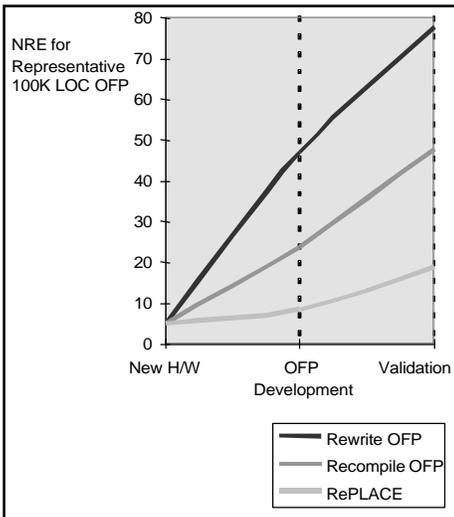


Figure 4: Comparison of Alternative Software Adaptions

- MH-60K Mission Computer (AP-102A).
- F-117 Mission Computer (AP-102A).

In all of these systems, legacy code performance improved from five to 20 times that of its legacy hardware environment. The cost and time to target a particular subsystem were a fraction of other approaches, including rewriting the OFP and custom chip replacements (hardware based emulation). The MIL-STD-1750A DISC has completed validation using the official acceptance test procedures and verification software originally developed and supported by the Systems Engineering Avionics Facility.

Comparisons of various alternatives to software adaptation are shown in Figure 4. The non-recurring costs for OFP development are, as would be expected, much higher for the case of an OFP rewrite. To a lesser extent, but still significant are the costs associated with an OFP rehost. This is because a rehost activity must still

address new machine dependencies and the immaturity of the associated software tools that are targeted for the new hardware. In addition, under the OFP rehost strategy, incremental software upgrades are difficult to implement. As the bottom curve illustrates, the cost for OFP development with a RePLACE approach is extremely small because the existing binary OFP code is used as is – unmodified. The costs included in the figure are representative of the time to tailor the I/O mapping software to support the new hardware and to incorporate legacy software tools into VIEWstation.

Code revalidation costs are significant for all three approaches. However, these costs are minimized with RePLACE since the structure of the embedded code is not changed, allowing the replacement computer to be retested at the black-box level using the existing functional qualification tests. TRW has developed a set of analytical tools to support this type of tradeoff for different user scenarios.

Upgrade Strategies

Potential upgrade strategies using the COTS software cover a wide spectrum of upgrade possibilities. These include the introduction of a new ruggedized COTS replacement box for an existing legacy avionics LRU. It also includes the introduction of a new microprocessor replacement module for insertion into an existing avionics LRU. Finally, included is a tool to mitigate the inherent risks associated with introducing both new hardware and software into a platform at the same time.

Figure 5 illustrates determining the baseline of new COTS-based hardware with the legacy OFP as a first step in the process of adding new functionality into an existing platform. Tailoring the

RePLACE DISC Legacy Instruction Set Architecture (ISA) is required to match the unique features and characteristics of the legacy machine; tailoring the VIEWstation is required to integrate with the legacy software tool-set.

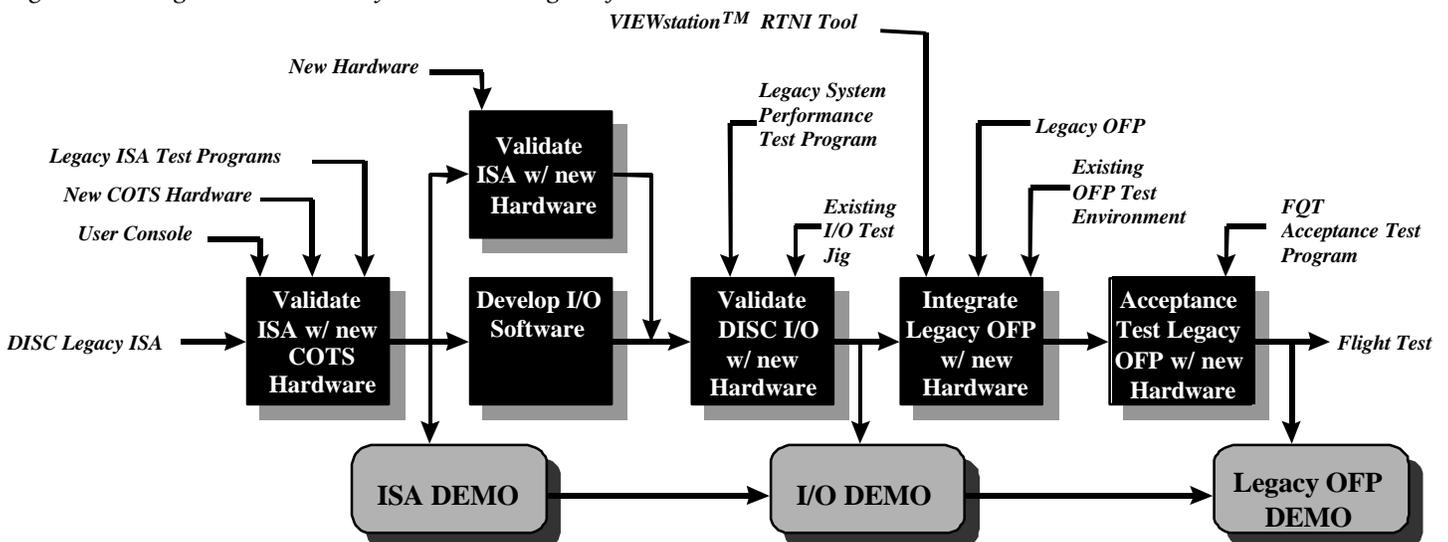
The validation steps include validating the execution of the ISA within the new microprocessor, validating the I/O characteristics using the new microprocessor, and integrating and running acceptance tests of the legacy OFP with the new hardware.

Conclusion

RePLACE technology offers the DoD customer significant supportability, producibility, and performance improvements through the introduction of COTS-based state-of-the-art hardware; it maintains backward compatibility with existing OFPs while providing the means to affordably migrate to state-of-the-art hardware and software technology. The result is significant cost savings, substantial risk reduction, and incremental upgrade opportunities for future enhancements. It also requires no custom hardware – it is an embedded software technology that leverages off the latest in commercial processor technology.

The engine can be supplied with a turnkey lab-quality or ruggedized box, or with a set of open system COTS board-level products. Also, PC-based configuration, control, and monitoring software is available to provide setup, maintenance, and user interaction with the engine. Although conceived for on-board avionics computer replacement strategies, it is equally effective to other embedded computer applications such as command and control systems, automated test equipment, weapon system trainers, and integrated support environments. ♦

Figure 5: Adding New Functionality into an Existing Platform



About the Authors



Jahn A. Luke is a senior program manager at the Embedded Information Systems Engineering Branch, Information Directorate, Air Force Research Laboratory (AFRL) where he is research and development task manager for the Computer Resources Support Improvement Program. He has more than 25 years of hardware and software experience at AFRL in the development of technologies for real-time simulation systems and embedded computer software. He is currently managing projects addressing aging avionics and legacy embedded information systems within the Department of Defense.

AFRL/IFTA
2241 Avionics Cir., Bldg. 620 Rm. S3-Y57
Wright Patterson AFB, OH 45433
Phone: (937) 785-6548 ext.3585
E-mail: jahn.luke@wpafb.af.mil



Douglas G. Haldeman is currently the project manager for TRW's Group II Mission Computer Replacement Program – a mission computer upgrade for the Navy's E-2C aircraft that uses RePLACE technology. Haldeman has a bachelor's of science degree in electrical engineering from Rose-Hulman Institute of Technology and a master's of science in electrical engineering from Purdue University. He has more than 25 years experience with TRW and has been working for the last 18 years in advanced avionics development and legacy avionics upgrades.

1900 Founders Drive Suite 202
Dayton, OH 45420
Phone: (937) 259-4970
Fax: (937) 259-4900
E-mail: doug.haldeman@trw.com



William J. Cannon is the manager of Legacy Processor Technologies for the TRW Dayton Avionics Engineering Center. He has over 26 years of experience in advanced embedded computer technology. Cannon's responsibilities include the assessment of legacy embedded computer systems upgrade opportunities and the technical oversight and direction of the development and application of the Reconfigurable Processor for Legacy Applications Code Execution (RePLACE) technology that is the subject of this paper. Cannon was the principle inventor of this technology.

1900 Founders Drive Suite 202
Dayton, OH 45420
Phone: (937) 259-4965
Fax: (937) 259-4900
E-mail: bill.cannon@trw.com

WEB SITES

Computer.org

<http://computer.org>

This site is the Institute of Electrical and Electronics Engineers (IEEE) Computer Society's technical and career resources Web site. The site features a search engine for the organization's 12 applications-oriented publications that foster active communication between practitioners and the research community, and its nine Transactions, research-oriented publications that document the state of the art in computer science. A search of "software legacy systems" turned up thousands of hits. The IEEE Computer Society is the world's leading organization of computer professionals with more than 100,000 members. The Computer Society is the largest of the 36 societies of IEEE. Its vision is to be the leading provider of technical information and services to the world's computing professionals. The society promotes an active exchange of information, ideas, and technological innovation among its members.

FCW.com

www.wargaming.net/Programming/69/Reengineering_index.htm

This is the Web site for Federal Computer Week magazine. Federal Computer Week is produced with emphasis on desktop, client/server, and enterprise-wide computing. A site search for Federal Computer Week turned up 226 results for legacy systems and legacy software articles.

GCN.com

www.gnc.com

This is the site for Government Computer News that reaches almost 90,000 government IT managers, delivering timely and insightful news coverage of the people and events that shape government technology.

Reengineering

www.wargaming.net/Programming/69/Reengineering_index.htm

This Web site features a list of reengineering books along with links for additional information on the publication. Topics include workflow reengineering, analyzing outsourcing, building enterprise information architectures, systems information reengineering, reengineering software, and more.

The RENAISSANCE Web

www.comp.lancs.ac.uk/projects/RenaissanceWeb/index.html

This site is intended as a resource for the entire software reengineering community. It also covers topics such as software maintenance, software evolution, reverse engineering, and software understanding. The site contains original content, taken from the ESPRIT RENAISSANCE software reengineering project, as well as a comprehensive list of other software reengineering resources on the Internet. The RENAISSANCE project is an ESPRIT funded research project into software reengineering and software evolution. One project result is a set of RENAISSANCE consultancy reports that covers architectural modeling, evolution risks economics, reverse engineering of system families, and the recycling of reusable code components.