MILITARY-COTS

Safety critical software

By Greg Rose

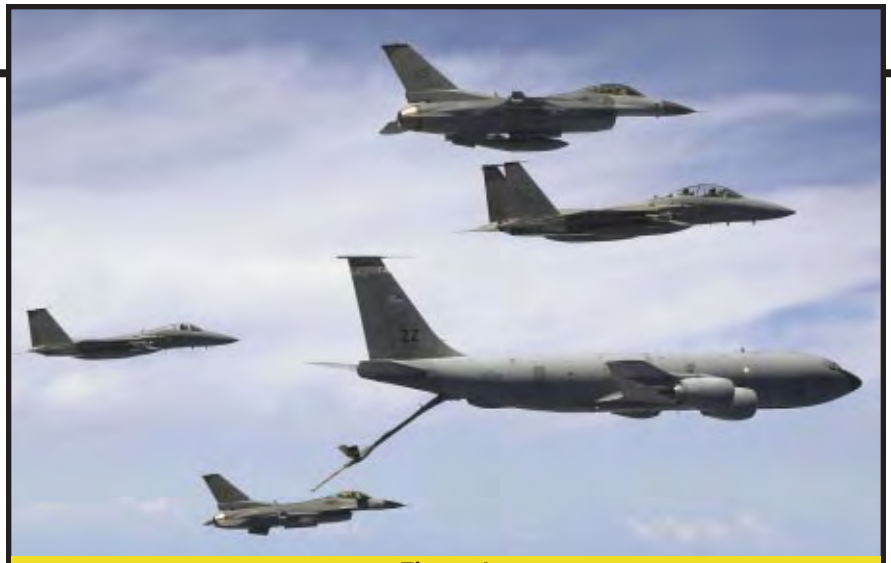

Figure 1

Software plays a critical role in almost every facet of our daily life – from cooking in our kitchens, to driving our cars, to working in our offices. Some of these systems are safety critical. Failure of software could cause catastrophic consequences for human life. Imagine the anti-lock brake system (ABS) in your car. A software failure here could render the ABS inoperable at a time when you need it most. For these types of safety-critical systems, having guidelines that define processes and objectives for the creation of software that focus on software quality, or the ability to use software that has been developed under this scrutiny, has tremendous value for developers of safety-critical systems.

In 1992, the Radio Technical Commission for Aeronautics (RTCA) approved the specification DO-178B for the aviation industry. This specification, the Software Considerations in Airborne Systems and Equipment Certification, was created to provide the aviation community guidance for determining, in a consistent manner, and with an acceptable level of confidence, that software aspects of airborne systems and equipment comply with airworthiness requirements. There are multiple levels of rigor that are applied to this software – from level E to level A – with level A being the most stringent. This guidance ties together system requirements, system life cycle processes, safety assessment processes and software life cycle processes, and documented traceability to show that the processes have been met. Historically, DO-178B was mandated for Air Transport class of aircraft and commercial avionics to comply with Federal Aviation Administration (FAA) regulations in safety critical systems. However in recent years, due to the Global Aviation Traffic Management (GATM) agreement which has international validity and applicability, airborne military (as shown in Figure 1) and space systems must also comply with DO-178B guidelines and certification for the safety of all aircraft.

The DO-178B specification does not contain anything magical; it enforces good software development practices and system design processes. It describes traceable processes for objectives such as:

- High-level requirements are developed
- Low-level requirements comply with high-level requirements
- Source code complies with low-level requirements
- Source code is traceable to low-level requirements
- Test coverage of high-level and low-level requirements is achieved

At higher levels, such as level A, these objectives must be verified by independent parties. No dead code is allowed in the system and all the requirements, code, and test information can be audited and traced. These documents are commonly called “artifacts.” For a piece of software to pass the rigor of DO-178B, satisfying the objectives of the specific level is required and there must be traceability through the artifacts to verify that the objectives have been met.

In contrast, typical commercial software is created and modified, and then due to time-to-market pressures or cost considerations, the developer may not choose to conduct independent reviews or testing of 100 percent of the code. Most commercial code is only reviewed or tested to an acceptable level of confidence to meet the business objectives of the manufacturer and for the criticality of the software. In many cases, this “typical” commercial software is not acceptable for use in systems where a malfunction of the software could lead to catastrophic consequences. This is where purchase and use of DO-178B verifiable software can give manufacturers confidence that they have used the highest quality software in their safety-critical application.

Commercial-off-the-shelf (COTS) operating systems are now available in the market, such as LynuxWorks’ LynxOS-178 (see Figure 2), that have been verified to DO-178B Level A. LynxOS-178 has been used in avionics subsystems by manufacturers such as Rockwell Collins, and is intended to be a common reusable element for safety-critical systems. In this case, all of the processes and objectives of DO-178B have been met for both the OS and the independent TCP/IP stack. The commercial availability of an operating system and TCP/IP stack enables manufacturers of DO-178B systems to get to market faster and lowers the overall business risks associated with the time and cost of certifying a system. And just as

importantly, it reduces the cost and time of certifying foundational elements, so manufacturers can concentrate on their value add, which is the application.

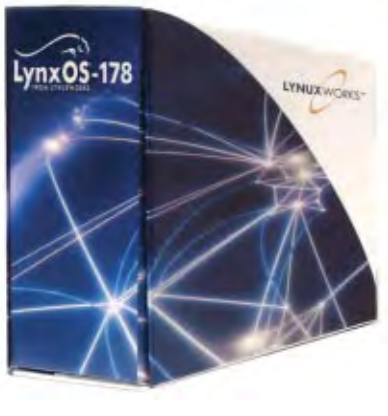

Figure 2

In addition to being a tremendous value to avionics manufacturers that must conform to DO-178B, availability of certifiable software is a tremendous value for all creators of all safety-critical systems. They can leverage these reliability benefits into their applications at a fraction of the historical cost. Commercial availability of certifiable software should spur development in markets for safety-critical applications and reduce time-to-market of important applications that can enhance the quality of our lives and protect us in critical situations.

Greg Rose *is the director of product management for LynuxWorks, Inc. Greg is a graduate of Iowa State University with a Bachelor of Science in Electrical Engineering. Prior to joining LynuxWorks in 1993, Greg had 11 years experience in embedded and real-time software design, and systems engineering. He has presented papers at Embedded Systems Conferences and published multiple articles in* EE Times *and other engineering trade publications.*

For more information, contact Greg at:

Greg Rose
LynuxWorks
855 Branham Lane East
San José, CA 95138-1018
Tel: 408-979-3900
Toll-free: 800-255-5969
Fax: 408-979-3920
Web site: www.lynuxworks.com