SOFTWARE DEFINED RADIO SOLUTIONS
NEW TECHNOLOGY AND JTRS PUSH THE ENVELOPE

Mark R. Turner (Harris Corporation, Rochester New York; e-mail: mturner@harris.com)

ABSTRACT

Software Defined Radio (SDR) solutions are being propelled forward through the evolution of several key technologies and by U.S. Government initiatives, most notably the Joint Tactical Radio System Program. This paper focuses on the role of Component Based Development (CBD) technology and the application of the Joint Tactical Radio System CBD based open systems architecture Software Communications Architecture on secure, battery-powered platforms for military use.

Harris Corporation is a multi-level participant in the Joint Tactical Radio System (JTRS) Program, including validation of the Software Communications Architecture (SCA) for battery-powered, Man-pack radio (MPR) platforms with a JTRS Step 2B contract. Results and observations from the Harris JTRS Step 2B Program are presented in this paper and also a description of current ongoing activities with the Step 2B extension contract. Harris Corporation is part of the winning JTRS Cluster 1 team for Ground/Vehicular and Airborne Systems. Harris is also actively involved in the pursuit of subsequent JTRS Clusters.

1. INTRODUCTION

Component Based Development (CBD) technology is already having a profound impact on the ultimate success of SDR initiatives, such as JTRS. The distinction between computers and radios are becoming blurred, i.e., is it a transmitter with a computer inside or a computer that transmits? The division between hardware and software continues to migrate at rapidly escalating rates. CBD technology has the potential to become the "industrial revolution" of software, promoting the advent of interchangeable software parts, built to predefined specifications. With respect to SDR solutions, the ability to reuse existing software components across multiple radio applications in an open framework, and the encapsulation of hardware specific capabilities and platform services through well-defined Application Programmer Interfaces (APIs) will facilitate true waveform portability both from practical application and affordability perspectives.

The challenge of multi-band / multi-channel RF technologies is exacerbated as the size of the platform shrinks to the man-portable domain with batteries as the primary power source. CBD technology can be used to support sophisticated application of software filtering and other interference management techniques, as well as dynamic spectrum management in accordance with local operational area requirements.

Information Security (INFOSEC) is a key performance parameter for military SDR platforms. Re-programmable INFOSEC is a key requirement for JTRS, intended to ensure that certified platforms can accept new cryptographic algorithms without hardware replacement. INFOSEC technology can benefit from the use of trusted, software components and applications, in conjunction with the controlled execution of programmable security policies. Special care must be taken using CBD technology to ensure that component connections do not violate security rules and policies across the red/black boundary of the system.

The embedment of networking technology inside the radio has enabled higher performance RF networking solutions, such as Harris' High Performance Waveform and Internet Protocol (IP) implementation in the AN/PRC-117F (C) MPR. CBD technology has the potential to continue this networking performance improvement through the logical interconnection of multiple traffic channels, while managing associated routing and cryptographic requirements.

2. JOINT TACTICAL RADIO SYSTEM

The objective of the Joint Tactical Radio System Program is to define and acquire a family of multi-mode, multi-band, programmable Software Defined Radios to increase operational flexibility, enhance Joint and Coalition interoperability, and reduce life cycle cost. These radio systems will provide network-centric capabilities and enable mission flexibility for the Department of Defense (DoD) Joint Vision 2020.

As stated in the JTRS Operational Requirements Document (ORD), the JTRS is required to provide interoperability across all geographical and organizational boundaries (both horizontal and vertical), so as to create an interoperable information transfer capability for Joint and Coalition operations. JTRS will be capable of transmitting voice, data, and video while operating in frequency bands from 2 to 2000 MHz (and beyond). By placing a premium on joint and coalition interoperability, the JTRS program focuses on solving interoperability issues currently with Services' legacy radios.

To facilitate the building of the JTRS family of radios, the JTRS Joint Program Office (JPO) tasked industry to develop and validate the SCA. The SCA consists of a set of rules and protocols, which define a Common Open Standards Architecture for SDR applications, making maximum use of commercial standards and APIs. This architecture supports implementation of SDR waveform applications and hardware that can be used in multiple operational environments.

3. COMPONENT BASED DEVELOPMENT

Component Based Development (CBD) technology is the foundation of the JTRS SCA and the cornerstone for the development and deployment of portable, reusable waveform applications. The creation of reusable software has long been a strongly desired, but unattainable objective of the Software Engineering community. Anticipated improvements in software reusability and reductions of software development costs have not been realized to date, particularly with regards to complex, custom-developed software systems, such as military radio and communications systems. The advents of Object-Oriented Programming techniques and Distributed Object-Oriented Computing have not eliminated this shortfall in expectations, as exemplified by the statement "software is not written, it's rewritten". CBD technology is based in the premise that a number of software systems elements reappear with sufficient regularity that these common parts can be developed once, rather than many times, and that systems can be assembled through reuse rather than rewritten over and over.

CBD has the potential to realize the reusability objectives of the Software Engineering community. CBD may perhaps even initiate the "industrial revolution" of software, promoting the advent of interchangeable software parts, built to predefined specifications. The fundamental premise of CBD can be thought of as a "buy, don't build" philosophy, promoted by Fred Brooks in [1].

For CBD, there are three widely recognized essential parts of a software component [2]:

1. <u>Interface</u>: Specifies to the component user (typically referred to as a consumer) what the component will do. Component interfaces must be documented in a complete and comprehensible manner, describing component behavior and protocols.
2. <u>Implementation</u>: The code that makes the component work. A component may be built with more than one implementation (i.e., one implementation that relies on a relational database and another that uses an object oriented database). Component implementation information should include a description of how it was constructed.
3. <u>Deployment</u>: The physical executable file used to make the component run. A component's deployment should describe the specific operating environment requirements.

In addition for CBD, software components should also support the following set of key characteristics:

1. Software components are <u>NOT</u> bound to specific applications, programming languages or platforms and are implementation approach neutral (i.e., procedural, object oriented or functional programming).
2. Software components are designed to pre-defined specifications.
3. Software components can be independently implemented.
4. Software components are delivered in encapsulated and replaceable containers, protecting the "what" from the "how".
5. When appropriate, software component behavior can be customized at configuration time.

With CBD, software components can be thought of as software integrated circuits (ICs) with a set of defined functionality, performance and input/output. Components can be assembled together to create entire applications, such as waveform applications for a Software Defined Radio. Figure 3.0-1 provides an example of a waveform application schematic depicting individual software components and associated connectivity.

In order to achieve true reusability, software components must be interchangeable. Application developers can substitute one component for another only if the interfaces are identical. For example, a waveform application using a 16 kbps frequency shift keying (FSK) modem can easily be converted to a 16 kbps waveform application using amplitude shift keying (ASK) by simply substituting an ASK modem component for a FSK

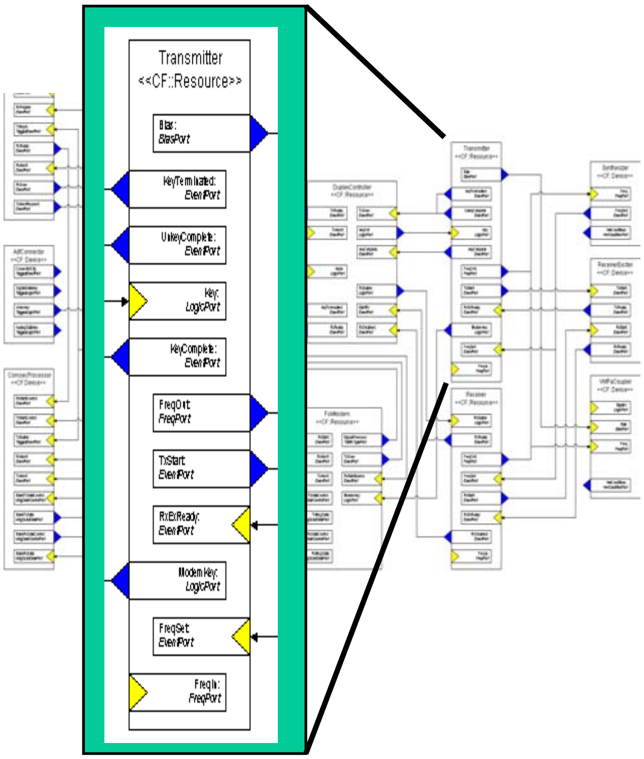

Figure 3.0-1 Waveform Application Schematic

modem component with identical interface definitions (i.e., behavior, protocol).

SDRs are highly diverse, ranging in size from large, high power vehicular, shipboard and fixed site systems, down to single channel, low power hand-held platforms. Clearly there are some basic radio technologies that are common across these diverse platforms. Operators use radios to communicate voice and/or data over-the-air. Operators feed information to and from the radio by connecting handsets and data equipment to the Input/Output (I/O) ports of the radio system. The information is typically digitized and processed within the radio. For example, the radio compresses the information, vocodes the audio, encrypts the data stream and modulates the information over the air via the RF components of the system. On one side of the system, all radios have I/O ports that operators connect to communication equipment. These interfaces define the edges of the system and are closest to the platform specific hardware. Information processing within the radio is application dependent. Since hardware interfaces are platform specific, and hence non-portable, service API definitions need to encapsulate these interfaces to facilitate waveform portability.

Consider the hypothetical waveform application depicted in Figure 3.0-2. The dotted lines in the diagram represent component interfaces with the platform Service Layer and the solid lines represent API interfaces. Note that this waveform application is entirely encapsulated by the platform Service Layer (typically provided by the platform manufacturer). This waveform application could be readily ported to another SDR provided the other radio supports the same Service Layer interfaces. The two platforms may have completely different underlying hardware architectures, but the Service Layer abstracts this out. True waveform portability can be achievable provided the edges of the system are well defined to encapsulate the underlying hardware.

Waveform Application development for a reusable JTRS Waveform Library is a key element of the JTRS Program strategy to ensure communications interoperability across a multitude of platforms and domains. It is important that these reusable waveform applications are developed in a manner that will support scalability from the smallest and least powerful platforms to the largest and most powerful platforms.

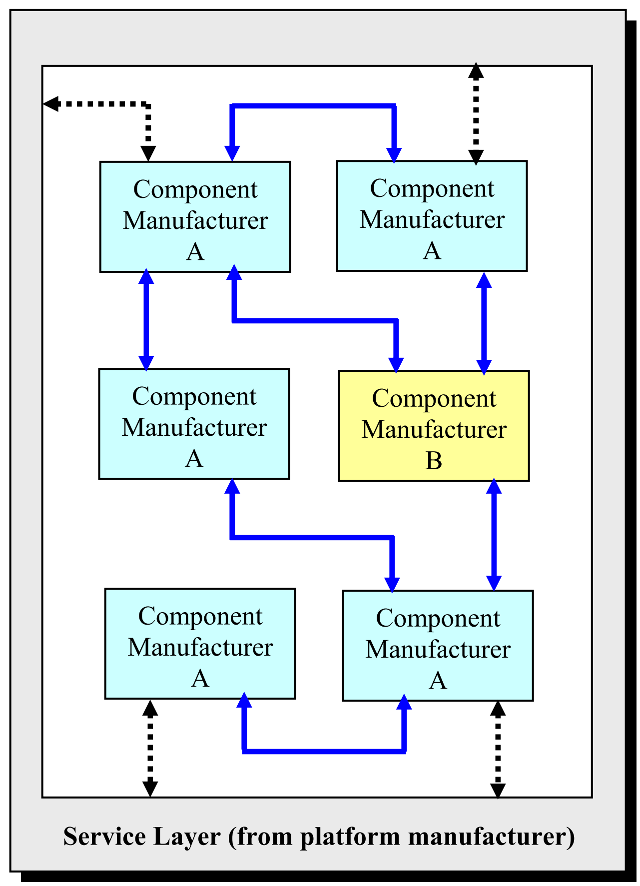

Figure 3.0-2 Example Waveform Application

The scope of the deployed software for SDRs clearly has grown during the evolution from structured programming, to object oriented programming, now moving to CBD. The level of software reuse and delivered quality has also grown during this evolution and is predicted to continue increasing as CBD becomes an integral part of SDR technology. Figure 3.0-3 provides an approximate estimation of SDR software deployment scope and reuse versus development methodology, presented in terms of 1000 Source Lines of Code (KSLOC) units.

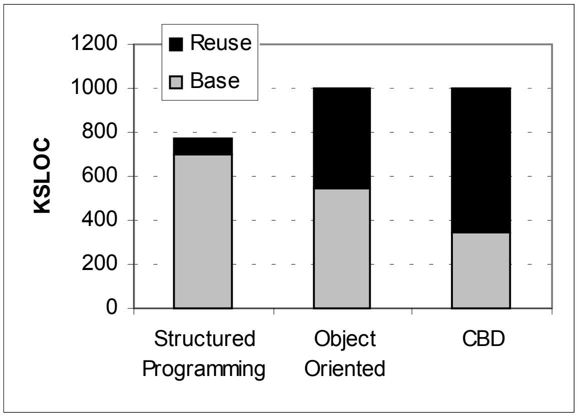

Figure 3.0-3 SDR Software Scope and Reuse

4. HARRIS JTRS STEP 2B PROGRAM

The objective of the JTRS Step 2B Program awarded to Harris Corporation in April 2000 is to validate the SCA for man-portable, battery powered platforms (specifically MPR form factors). While these platforms are constrained by size, weight, and power, they often require very high performance for critical system characteristics such as, system turn-on time and system reconfiguration time. Because of this, man-portable, battery powered platforms have unique requirements when compared to larger vehicular, airborne and fixed station platforms. The number and location of processing elements, memory storage capacity and power control of hardware elements are key design considerations when developing a battery-powered platform.

4.1. Initial Step 2B Program

Harris performed this validation effort by building the JTRS Manpack Test-bed Radio (JMTR) platform, which is based on the AN/PRC-117F (C) MPR. Refer to Figure 4.1-1 for a picture depicting a JMTR.

The AN/PRC-117F MPR is part of the Harris Falcon II™ radio family, which is an advanced tactical suite of SDR products that includes HF, VHF, UHF and multi-band radios for US Government and international applications. As is the case with the current AN/PRC-117F (C) MPR, the JMTR provides continuous, multiband frequency spectrum coverage from 30 MHz to 512 MHz and supports multi-mode operations for both Line-Of-Sight (LOS) and UHF Tactical Satellite (TACSAT) missions.

Figure 4.1-1 JTRS Man-pack Test-bed Radio (JMTR)

As part of the JTRS Step 2B program, Harris developed an SCA V1.1 Core Framework and a set of Waveform Applications that were installed on the JMTR and demonstrated to the JTRS JPO. A series of validation tests were also executed in order to measure and analyze the performance of the JMTR, as well as compare it to today's legacy systems.

Harris' Core Framework was installed on a black-side General Purpose Processor (GPP). Harris developed several Core Framework components used in conjunction with the following third party software:

- Wind River's VxWorks V5.4 for the real-time, POSIX compliant operating system.
- Objective Interface System's ORBexpress V2.3.1 for the CORBA Object Request Broker (ORB).
- Exigent (Harris) Domain Manager Run-Time Environment (DMRE) for the Domain Management portion of the Core Framework.

For the CORBA ORB selected by Harris, an embeddable naming service was not available at the time. This required the use of a CORBA naming service on a personal computer that is also accessed via the radio's Ethernet interface. Other than the CORBA naming service, all of the Core Framework software executes on the JMTR hardware.

Harris developed two waveform applications for the validation. The first waveform application developed was Plain-Text (PT) Continuously Variable/Slope Delta

(CVSD) Voice. This Line-Of-Sight (LOS) waveform application is equivalent to an unencrypted version of KY-57 compatible voice. It uses a 16kbps Frequency Shift Key (FSK) modem that operates over the 30 to 90 MHz frequency range.

For the second waveform application, Harris developed a portion of Mil-Std-188-181B for UHF Dedicated SATCOM operation. This included PT data operation at rates of 4800bps, 19.2kbps, 32kbps, 38.4kbps and 48kbps using Continuous Phase Modulation (CPM). This provided a much more processing intensive waveform application for the validation effort.

Harris completed the initial JTRS Step 2B Program in October 2001, on–time and within budget. **In April 2001, Harris completed the first ever over-the-air JTRS waveform demonstration.** This demonstration included successful LOS digital voice communications between a JMTR running the Harris JTRS software and a legacy AN/PRC-117F (C) MPR. Also in October 2001, Harris demonstrated successful data communications with the AN/PRC-117F (C) MPR using the Mil-Std-188-181B waveform over a UHF TACSAT channel simulator.

4.2. Step 2B Results and Observations

The Harris Step 2B activity identified several areas of the SCA that significantly stressed the JMTR hardware platform and operational usability, including:

- CPU processing requirements: radio power-up time, application switch time, parameter change time and TX/RX turnaround time.
- Non-volatile memory (ROM/Flash) and volatile memory (RAM) requirements.
- Data latency increases due to transport of data via packets in a CORBA environment.

The following tables provide performance data collected during the Harris Step 2B Program; note that this data does not include any INFOSEC overhead, since SCA V1.1 did not address INFOSEC requirements. Table 6.2-1 provides a timing performance comparison between the JMTR running the Harris Step 2B JTRS software and a legacy AN/PRC-117F (C) radio. Note that the measurement data has been normalized to the performance of the legacy radio. Table 6.2-2 provides a memory storage capacity comparison between the JMTR running the Harris Step 2B JTRS software and a legacy AN/PRC-117F (C) radio. Note that the measurement data has been normalized to the memory capacities of the legacy radio and adjusted to reflect the full waveform capabilities of the AN/PRC-117F (C) radio. Table 6.2-3 indicates the percent of memory utilization by the SCA Operating Environment and developed waveform applications. The scope of these waveform applications will clearly be somewhat less than final JTRS library waveform applications (i.e. with the addition of security capabilities), but does represent the impact of the JTRS Operating Environment.

Table 6.2-1 Timing Performance Comparisons

Measurement	JTRS	Legacy Radio
Power-up	8	1
Application switch time	16	1
Parameter change time	4	1
TX/RX turnaround time	3	1

Table 6.2-2 Memory Storage Comparison

Measurement	JTRS	Legacy Radio
ROM	10.0	1
RAM	8.0	1

Table 6.2-3 Memory Utilization

Measurement	JTRS
Operating Environment	76%
Voice waveform application	12%
Data waveform application	11%
Total	100%

4.3. Step 2B Program Extension

Harris Corporation was awarded an extension to the Step 2B contract in October 2001 (referred to as 2BX) to conduct further SCA validation on a battery-powered, MPR platform. The Step 2BX Program was motivated by several circumstances, including the following:

- Significant evolution of the SCA from V1.1 to V2.2.
- Insufficient validation of the SCA Security Supplement due to its delayed release.
- Results of the initial 2B contract identified areas of "SCA induced" stress on MPR operational performance warranting further validation.

There are two main objectives of the Step 2BX contract: validate SCA Security and API supplements, develop and test an SCA V2.2 Lightweight Core Framework (CF-Lite) on a MPR platform. As part of the Harris contribution to the Step 2BX Program, the JMTR hardware platforms delivered under the initial Step 2B Program will be upgraded with increased processing and memory storage capacities, in conjunction with the incorporation of a programmable INFOSEC module using a Harris Sierra cryptographic module (refer to Figure 6.3-1).

Figure 6.3-1
Sierra Module

INFOSEC is a key component to the success of the JTRS Program, including the development and deployment of a programmable INFOSEC module. Without a programmable INFOSEC module, the flexibility and growth capability of a JTRS system will be limited or potentially require expensive hardware upgrades. The rapid evolution of the Internet and wireless networking is dictating the development of next-generation cryptography. The development and integration of programmable cryptographic modules will allow JTRS-compliant systems to evolve as new forms of cryptography are developed.

For the JTRS Step 2BX Program contract extension, Harris has selected the following third party software:

- QNX V6.2 for the real-time, POSIX compliant operating system.
- Objective Interface System's ORBexpress V2.5 for the CORBA Object Request Broker (ORB).

Harris is in the process of updating the JTRS waveform applications developed under the initial Step 2B Program to be compliant with SCA V2.2, including the SCA Security and API Supplements. As part of the JTRS Step 2BX Program, Harris will also specify, design and implement an SCA V2.2 Core Framework that is optimized for battery-powered platforms, referred to as CF-Lite. These optimizations are intended to conserve platform processing power and storage requirements, reduce system start-up and net switching times to usable levels, while maintaining the integrity of waveform application interfaces (and therefore ensuring waveform portability). The CF-Lite concept is analogous to the PC paradigm, where different levels of Operating Environments are targeted for different sized platforms, while maintaining the portability of applications. Based on the results of the Step 2BX Program, Harris will submit a set of recommendations to the JPO for consideration and incorporation into future SCA revisions.

5. CONCLUSIONS

New technology and the U.S. DoD JTRS Program are pushing the envelope for SDR solutions. Component Based Development is a key technology for development and deployment of complex software applications for SDR platforms. CBD is the foundation for the JTRS SCA. The application of the SCA and CBD technology to battery-powered platforms has special implications for platform performance, security and networking capabilities.

Waveform application portability is a key tenant of the JTRS Program for the express purpose of ensuring communications interoperability across increasingly more complex missions. Waveform applications developed for the JTRS Waveform Library need to be validated on a platform that represents the smallest platform domain. This validation should begin during the requirements analysis and design phases of the Waveform Application development to limit the risk of significant re-work during the testing and certification phases.

Harris has been a pioneer in the military SDR industry, building and fielding tactical SDRs covering 2 – 512 MHz spectrum, since the late 1980's. Harris is a multi-level JTRS participant, from a successful, ongoing JTRS Step 2B program, to providing the programmable INFOSEC solution for the JTRS Step 2C program, to being a member of the winning JTRS Cluster 1 Team, to being a well positioned industry leader for small, battery-powered platforms with scalable, programmable INFOSEC. Harris is providing real SDR and JTRS solutions for the warfighter today.

6. REFERENCES

[1] Brooks, F. P. Jr. "No Silver Bullet: Essence and Accidents of Software Engineering", Computer 20, April 1987

[2] Szyperski, Clemens, "Component Software – Beyond Object Oriented Programming", Addison-Wesley / ACM Press 1998.

[3] "Joint Tactical Radio System – Operational Requirements Document", Revision 2.3, 23 November 2001.

[4] MSRC-5000SCA: "Software Communications Architecture", Version 2.2, November 17, 2001.

