CERTIFICATION OF ESRA-COMPLIANT SOFTWARE DEFINED RADIOS

Stefan Nagel*, Volker Blaschke*, Jens Elsner*, Friedrich K. Jondral*,
Dimitrios Symeonidis+
*) Institut für Nachrichtentechnik, Universität Karlsruhe (TH), Germany
{nagel;blaschke;elsner;fj}@int.uni-karlsruhe.de
+) JRC Joint Research Centre – European Commission, Ispra, Italy
dimitrios.symeonidis@jrc.it

Abstract[1] section: 6.6 Waveform Portability, 17.1 SDR Performance Certification and Accreditation

1. INTRODUCTION

In this paper we attempt to tackle the technical requirements for the certification of SDR and its components. We adhere to the platform-waveform paradigm followed by all major SDR Architectures (JTRS SCA, OMG SWRadio, ESRA) and attempt to identify the necessary tests to assure seamless interoperability and to facilitate waveform portability. We emphasize the connection between certification and standardization and discuss the benefits of a reference implementation. Apart from API availability and correct implementation testing for the platform and the waveform, we suggest the testing of non-functional requirements such as performance. Finally, we acknowledge the delicate issue of security certification.

2. CERTIFICATION PROCESS IN SDR SYSTEMS

Certification of SDR platforms and waveforms is necessary to facilitate portability of waveforms between platforms by verifying the interfaces between them closely follow the ESRA standard. This in turn avoids duplication of effort and increases competition by allowing SDR components for platforms or waveforms developed independently to integrate with minimum porting effort, thus encouraging innovation and lowering the market entry barrier.

Developing a certification process is per se a challenging goal; developing an SDR certification process is even more challenging because Software Defined Radio is a technology that involves two very different disciplines: software and radio. Each of these disciplines is very wide, mature and well studied; however the cross-section of both, which is Software Defined Radio, is a new technology that is not yet as mature as each one of them individually.

[1] The authors agree that the paper may be reviewed by the SDR Forum paper evaluation committee for purposes of inclusion in the conference. The authors allow the inclusion of the paper in SDR Forum conference publications, and other SDR Forum publications, if the paper is accepted.

3. PLATFORM CERTIFICATION

Picking up from the goals and requirements described in the previous paragraph, the certification of an SDR platform includes first of all the verification of the APIs each of its components advertises to other platform components or to waveform application components. This verification includes not only their existence, but also their correct behavior.

Furthermore, tests targeted at the entire platform (instead of at each component individually) verify the correct integration of the platform as a whole, meaning for example - but not only- that the components are installed, instantiated and connected to each other in the intended collaborative way.

Another important parameter to be tested and certified is the performance of a platform to execute the tasks required in a dependable, reliable and timely manner. The speed and reliability of the reconfiguration procedure (changing waveform parameters on-the-fly, loading/unloading waveforms) is important in the SDR world and even more so in the Cognitive Radio (CR) world. Besides, the metrics used by platform components to advertise their performance capabilities need to be matched with the performance requirements for deployment of each waveform component.

4. WAVEFORM CERTIFICATION

Waveform certification deals with certification for interoperability and portability. Interoperability is easily verified by testing compatibility with legacy devices. In SDR development however, interoperability is affected by both, platform and waveform code. E.g., a waveform might by be interoperable with legacy radios on one specific SDR platform, but might fail on another. This shows that the problems of interoperability and portability are closely connected. Interoperability can not be guaranteed for a waveform alone, but only for the package of waveform and platform.

As opposed to interoperability, portability is difficult to measure. In an ideal homogeneous environment, the waveform code of one platform would work without modifications on a different SDR. In analogy to general-purpose application programming in, e.g., Java, one might think of a common SDR virtual machine definition to guarantee full portability. In reality, however, this is unfortunately not possible due to high processing data rates, delay and throughput requirements of an SDR. SDR platform developers tackle these problems differently, depending on complexity requirements of the specific task at hand. This results in a variety of different SDR platform architectures available. Waveform code running on one platform will certainly not work without modifications on another. The more similar the architectures, the less complex the porting effort is likely to be. Quantifying the costs, and hence "measuring portability", however, is difficult.

Rather than certifying code for portability after it is created, it is more sensible to adhere to development standards, which implicitly guarantee a certain measure of code portability. To achieve this, the WINTSEC portability development standards are based on the Model Driven Architecture (MDA) of the Object Management Group (OMG) [1].

The OMG specification – as recommended by ESRA – promotes the use of the MDA to produce platform-independent (PIM) and platform-specific (PSM) models applicable to SDR. A PIM is a high-level representation of the waveform, without referencing to a certain hardware architecture. It could be a Matlab™ code representation for signal processing or a UML model for protocol design. The next step towards code generation is then the definition of a model for a waveform on a specific platform: the PSM.

WINTSEC proposes an additional approach to facilitate waveform portability that is complimentary to the MDA approach: to specify an API between waveform components for a given wireless standard to be implemented. By a-priori defining this API, code porting is simplified at the expense of implementation flexibility.

The full paper will examine the development process and API specifications by implementing an example waveform according to the MDA, leading to a portable waveform.

5. REFERENCES

[1] Model Driven Architecture (MDA), Architecture Board ORMSC, Object Management Group, July 2001.