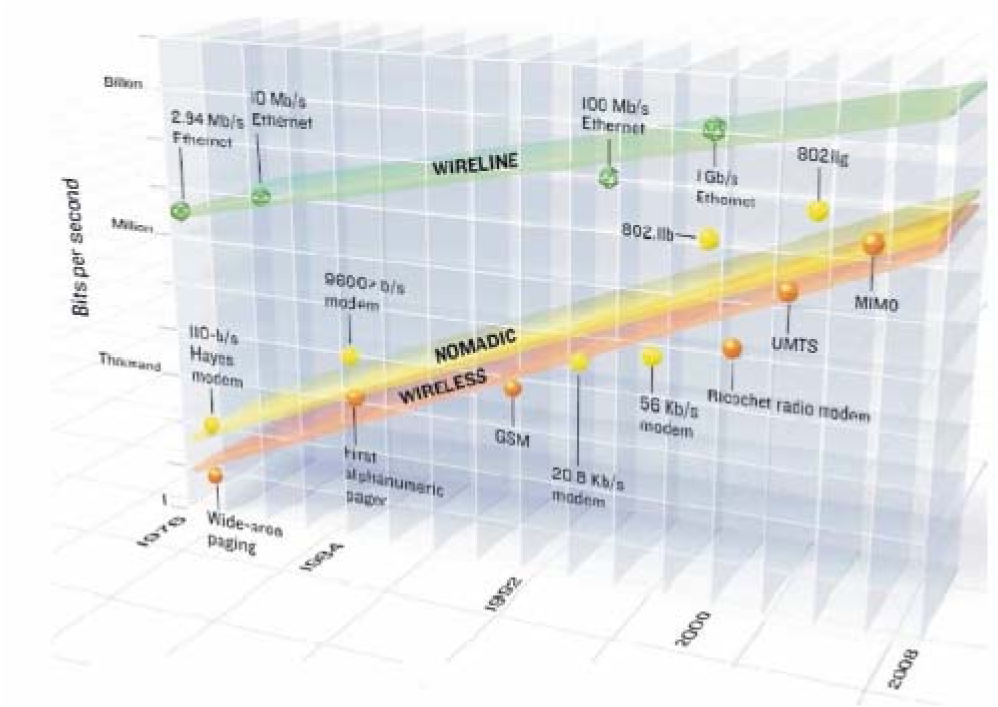

Figure 1 Capacity of wireless offerings will soon compare with wireline. Figure taken from IEEE Spectrum, Vol.41, no.7 (NA), July 2004, p. 58.

Introduction:

Advances in technology are creating the potential for wireless systems to use radio spectrum more intensively and more efficiently than in the past (see Figure 1). These rapid developments are being driven by the high level of connectivity demanded by the market. People would like to have the same broadband capabilities they do at their PCs, while maintaining the wireless freedom that their cell phone provides. Unfortunately, **conflicting standards** and **soaring deployment costs** have limited wide scale adoption of next generation wireless/cellular technologies. What is needed is a solution capable of resolving both the standards issue, as well as a method to reduce the soaring cost of research and development. Software Defined Radio technology is finally poised in a position to offer these solutions.

Research into the feasibility and development of a software controlled reconfigurable radio system has been active for the last 10 years[i]. The key idea of software radio is to **perform all radio functionality in software algorithms running on general purpose processors** (GPPs), rather than on the dedicated hardware components that we are familiar with today. Since GPPs can essentially run 'any' program, software radios themselves are imbued with inherent flexibility. Particular standards are no longer an issue since the device may easily be reconfigured to operate with whatever standard the user desires. Consider an orbiting satellite. If it is software radio based, then as new broadcast standards arise such as the evolution from DVB-S[ii] to DVB-S2[iii], then the software algorithm defining the new standard may be uploaded to the satellite. The satellite may now reconfigure itself to operate using this new standard. This is remarkable in considering that at the time the satellite was deployed, that particular standard might not even have existed!

Software Defined Radio – Defined

The FCC defines a device as a software defined radio device to include[iv] *any radio that includes a transmitter in which the operating parameters of frequency range, modulation type or maximum output power can be altered by making a change in software without making any changes to hardware components that affect the radio frequency emissions.*

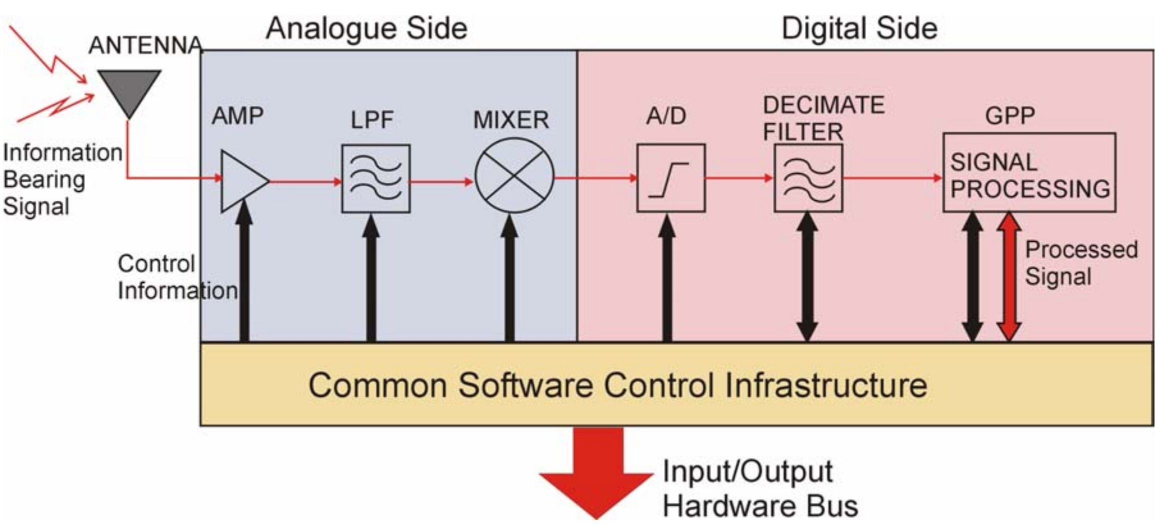

Figure 2 Basic configuration of a Software Radio.

In Figure 2 the basic configuration of a software defined radio is shown. Particular implementations will vary, but all software defined radios share three common sections: (1) Analogue Side, (2) Digital Side, and (3) Software Control of the device. The ultimate goal of SDR is to remove the Analogue side entirely by performing the digitization (A/D) immediately after the antenna. Unfortunately, this is beyond current technology so the signal is first converted to an intermediate frequency and then digitized. All subsequent signal processing is performed digitally.

The real advantage of SDRs is in the nature of the digital processing. Not only is digital filtering/processing superior to analogue processing, but owing to the reprogramability of the digital signal processing devices, they themselves may be reconfigured to perform different kinds of signal processing. Consider a cellular phone operating on a network using GSM. If that device now moves to a network employing an IS-95 or EDGE standard, then that phone will not be able to function on the new network. However, if it is able to reconfigure its components to function using these standards, then the device will be able to function. Thus by their inherent nature software radios have the potential to solve one the key wireless stumbling blocks (namely conflicting standards).

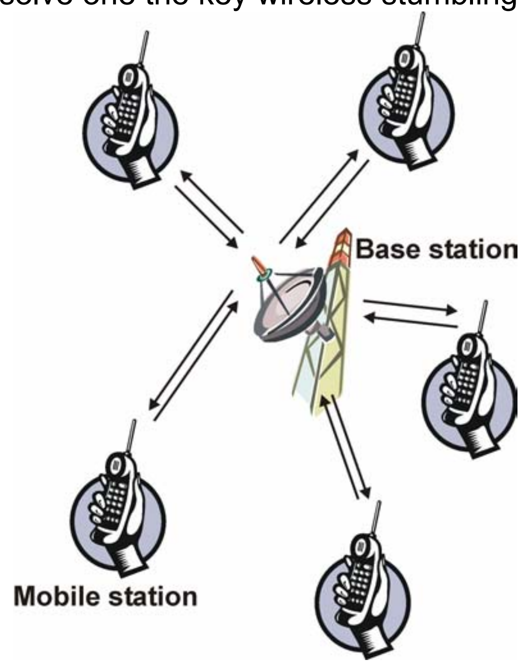

Figure 3 Hub network where a single base station controls traffic flow.

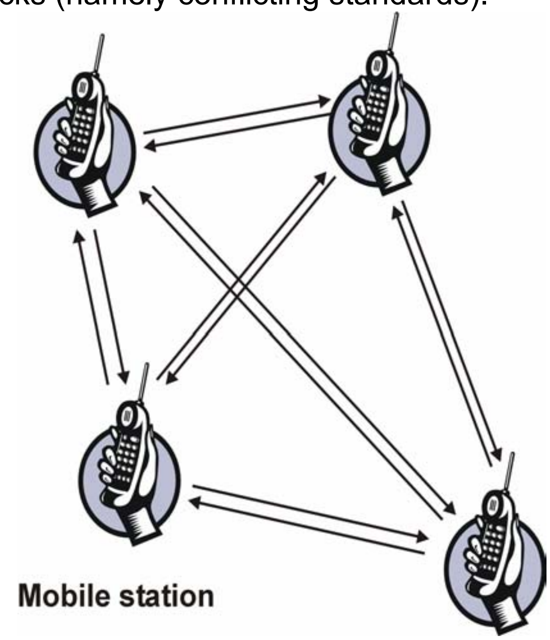

Figure 4 Mesh network where no single controlling device exists.

It is important to realize that the SDR concept is more profound than simply offering reconfigurable phones or other communications devices. The SDR framework itself is *independent of underlying radio network technologies and* *services*. SDR technology allows different *kinds* of networks to be developed. Current network topology is generally determined by the 'base stations', i.e., the communications devices that are used to control the wireless traffic flow in their area. Consider Figure 3 which illustrates a typical wireless cell. All traffic flow is routed through the base station. This networking topology is referred to a hub network. If two users wish to communicate they must go through the base station. Conversely, if the mobiles were to receive a software upgrade defining a Mesh network standard (as is shown in Figure 4) then no base station is needed to facilitate communications. The entire network topology has changed with only a simple software upgrade. This reconfigurability could be enormously useful if the devices are deployed in geographic locations where no present infrastructure exists.

Software Engineering

Past communications devices were generally developed by a single provider using their own proprietary solutions (both hardware and software). This meant however, that devices from different venders were unable to communicate with one another. Note that software radios do not inherently solve this problem either since individual vendors will use their own proprietary software to manage the device. The US military, a large user of communication devices, found itself suffering from this very problem – it owned many communications devices from multiple providers that were unable to communicate with each other. This proved to be extremely inefficient and costly. As a result it developed a new set of standards that all its future communications devices must adhere to. This set of standards is collectively referred to as the Software Communication Architecture - Core Framework (SCA-CF)[1].

The SCA-CF is an architectural concept defining the essential, "core" set of open software interfaces and profiles that provide for the deployment, management, interconnection, and intercommunication of software application components in a communications system[v]. Hardware components are "abstracted" using object orientated concepts as virtual software devices. This enables them to be more easily controlled though software interaction. The goal of the CF is

> *to ensure the portability and reconfigurability of the software and hardware, and to ensure interoperability of devices compliant to the SCA-CF.*

[1] The Software Communication Architecture specification is published by the Joint Tactical Radio System (JTRS) Joint Program Office (JPO). See http://jtrs.army.mil/ for further details.

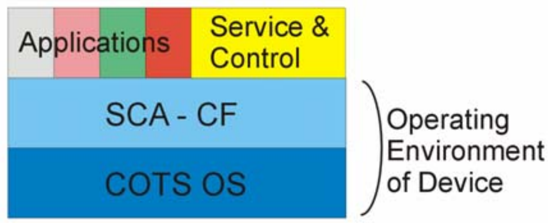

Figure 5 Levels of software engineering required for development of a SDR.

Figure 5 shows the four primary levels of software engineering required for the development of a software defined radio. At the lowest level is the Commercial-Off-the-Shelf (COTS) operating system (OS). This is required by the CF to be a (mostly) POSIX compliant OS such as Linux. At the highest level is the software that is responsible for maintaining service and control. This would include for example the drivers necessary for operating the mixer or the decimation filters in the radio chain (see Figure 2). Also found at this level are the applications that we would like our radio to run. A particular application may be the IS-95 'waveform'[2], or perhaps the ability to access a WiFi hot spot. The algorithms responsible for executing the signal processing necessary for the implementation of the waveform are run on the GPP (Figure 2).

Not surprisingly the SCA-CF is 'sandwiched' between the OS and the Service & Control sections where is it able to control the method of the flow of information (the nature of the actual data itself is irrelevant). It is this ability to control the method of operation that makes it possible for two completely different radio devices to communicate so long as they are both operating according to the set of standards outlined in the SCA-CF.

Thus the software radio paradigm coupled with the SCA-CF is an implementation that solves the standards and interoperability issue. Additionally, it also dramatically reduces development costs since it frees companies of developing the entire radio complete with software. So long as the individual hardware and software components are SCA compliant, the device will operate properly. Companies can now specialize in developing particular segments of the radio corresponding to what they are best at. For example, software companies specializing in signal processing solutions may develop the waveforms. This provides a better product, as well as saves money since presumably they have already invested in their current infrastructure. It is really just a matter of time before all communications devices are software controlled communications devices.

---

[2] A radio waveform is the implementation of a particular standard or application that we wish to run on our communications device.

Prospects/Applications for Technology

Array views the SDR paradigm as the natural evolution of telecommunication enabling technology. SDR's promise offers:

i. Increased flexibility and interoperability of communication systems.
ii. Upgradeability in terms of ease of both hardware and software technology insertion.
iii. Reduced system development, operation, and supportability costs.
iv. By altering the wireless value chain, SME companies may now play a more pronounced role in offering telecommunications solutions.
v. Advanced networking capabilities to allow truly "portable" networks.
vi. Uniform communication across commercial, civil, federal and military organizations.

Array will leverage these technologies to situate itself as an SDR system integrator and application developer. Array will develop target hardware and software solutions for SDR based platforms compliant to the SCA-CF. COTS components (both hardware and software) that are SCA compliant can then be guaranteed to function properly when incorporated into an SDR's architecture. For example, assume a new standard is developed for a particular communications market segment. Provided the waveform associated with the new standard is made SCA compliant it can seamlessly be integrated into the radio and now offer the enhanced performance of the new standard.

Benefits of this approach:

Recent research suggests that one of the biggest problems based on a COTS approach is failure of standardization of the individual components. Forcing standardization by developing SCA-CF compliant components will greatly simplify the SDR solution. In the long run standardized products provide cost savings and offer greater reliability.

Providing standard interfaces significantly increases the ability of the device to be reconfigurable and therefore more flexible. In order for SDR to become truly successful, it must be interoperable. This can only reliably happen if established standards exist detailing how the interaction should take place.

Easy upgrades. The ability to upgrade allows network providers/end users to increase the products' life cycle. Upgrades may take a reasonable phased approach as operators will only need to upgrade particular components. Future upgrades will be dictated by the market. A low-risk 'wait and see' approach can be used. This is significantly cheaper than redeveloping the entire network architecture as was necessary for 3G.

Application/Waveform development. Leveraging commercial standards to reduce development time offers cost savings. Additionally, as standard application libraries become more developed, the development time of new applications are reduced through the ability to reuse previous design modules.

References:

[i] Mitola, J., “The Software Radio Architecture,”, *IEEE Communications Magazine*, May, 1995.

[ii] Digital Video Broadcasting (DVB), European Standard (Telecommunications series) EN 300 421, V1.1.2, 1997-08.

[iii] Digital Video Broadcasting (DVB), Second Generation, European Standard (Telecommunications series) ETSI EN 302 307, V1.1.1, 2004-06.

[iv] Facilitating Opportunities for Flexible, Efficient, and Reliable Spectrum Use Employing Cognitive Radio Technologies. Document: FCC 05-57, 2005-03.

[v] Software Communications Architecture, SCA V3.0, http://jtrs.army.mil/sections/technicalinformation/fset_technical_sca.html, Document: JTRS-5000SCA rev.3, 2004-08.