

## Field trials of an all-software GSM base station

The cellular industry's first-ever commercial deployment of software-defined radio (SDR) had some system design and integration challenges as well as real and potential outcomes.

By Jeffrey Steinheider

As part of the Mid-Tex Cellular radio access network, Vanu, Cambridge, Mass., installed a software radio GSM base station in DeLeon, Texas, in June 2003. The base-station platform comprised an off-the-shelf HP ProLiant server with two 2.8 GHz Intel processors running Linux and connected to an ADC Digivance radio transceiver. All of the signal processing, protocol processing and GSM BSC functionality were implemented as application software running on the HP server. Mid-Tex operates an IS-136 AMPS/TDMA system covering 8000 square miles in six counties, about two hours west of Dallas-Fort Worth airport.

### Why use software radio?

Two fundamental design decisions give software radio flexibility and portability. First, no FPGAs, ASICs or DSPs are used for application-specific signal processing in the system. Second, all software is written in high-level, object-oriented code and built to run on any reasonably POSIX-compliant operating system.

By using off-the-shelf hardware and commercial processing platforms, these radio systems leverage the economic benefits of high-volume components and the technology advances driven by Moore's Law. Additionally, as new, faster processors are introduced, they can be integrated into new products quickly and at low cost without expensive software redesign and development.

The Vanu solutions not only define, but also perform all signal processing using software applications. The advantage of moving signal processing into software is increased flexibility, allowing for maximum software reuse across platforms and hardware generations. In addition to reducing software devel-

opment costs, this method allows hardware components to be upgraded without incurring the high cost of software redevelopment, effectively tracking Moore's Law.

Vanu software radio also supports rapid advancement of wireless solutions. Wireless manufacturers will no longer need to deploy new hardware to keep pace with leading-edge technologies and standards. The software radio devices download software, thus reducing operating costs and increasing revenue opportunities for their owners.

### How does Vanu software radio work?

Software radio architecture consists of three basic blocks: the antenna subsystem, RF conversion and processing platform as shown in Figure 1. The antenna subsystem includes components such as a low-noise amplifier (LNA) on receive and a multicarrier power amplifier (MCPA) on transmit. A 30 W MCPA was used for the Mid-Tex trial. The RF conversion section is a block up/downconverter that includes analog-to-digital and digital-to-analog conversion. The RF conversion for the Mid-Tex trial was realized using the ADC Digivance 850 MHz Long Range Coverage System, which performed block up/down conversion of the 25 MHz cellular band. Finally, the processing platform was the HP ProLiant server, which supports four GSM TRXs (32 traffic channels) per server. No waveform-specific processing occurs in hardware; all of the signal processing occurs in application-level software running on the server. Once the wireless signal is moved into processor memory, an operating system layer and two stacks of application-level software layers perform the signal processing. Device drivers are in the operating

system that allow the applications to control and monitor hardware components. The hardware components, operating system and device drivers compose the software radio platform. Regardless of the hardware implementation choices for each section, the basic system architecture remains the same.

### The base station trial

The trial installation consisted of two base station transceivers (BTS) and a base station controller (BSC), with each running on an industry-standard HP ProLiant DL380 server with dual Intel Xeon 2.8 GHz processors. All of the signal processing, protocol processing, and BSC functionality was implemented as application-level software running on top of the Linux operating system (Figure 2). The BTS systems used the ADC Digivance Long Range Coverage Solution as an RF interface.

Each BTS provided two TRXs of GSM capacity in one sector, which can be upgraded to four TRXs each, via a software-only upgrade in the future. Each TRX contained eight time slots. Two of the time slots per BTS were used for the control channel, so each BTS supported 14 simultaneous voice calls. The first BTS was located in DeLeon, while the second BTS was located 11 miles away in the neighboring town of Gorman. The BSC equipment was located in the DeLeon central office. One of the goals for the trial was to evaluate handover between the two BTSs, so directional antennas were used to ensure overlapping coverage on the road between the two towers. The current Mid-Tex IS-136 system that operates from the same towers uses omnidirectional antennas and does not provide continuous coverage between the two towns.

The ADC Digivance system provides a fiber-optic link for remote antennas. This capability was used for the Gorman BTS installation. The HP server running the BTS software was co-located in the DeLeon central office with the BSC, with its antenna 11 miles away.

Co-location of several base stations in a central location is known as base-station hoteling. Hoteling offers cost savings to an operator because it reduces footprint and eliminates climate control at the cell tower,

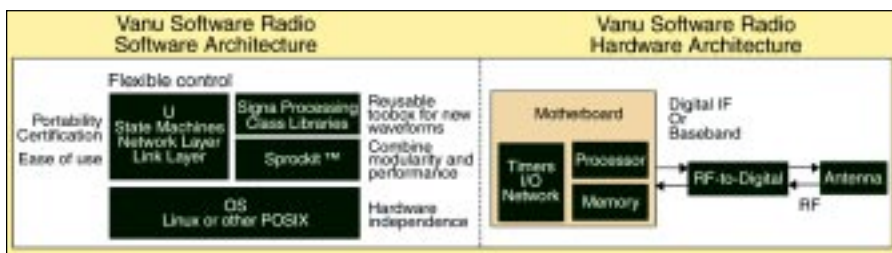


Figure 1. Vanu software radio general hardware and software architecture.

while centralizing equipment, which reduces maintenance costs.

For the DeLeon BTS, the BTS server was installed at the tower site. It was connected via a T1 link to the BSC server. This was a more traditional cellular system installation compared to the Gorman BTS. An HP Procurve gigabit Ethernet switch at the DeLeon central office connected the BSC server, the Gorman BTS server and the Cisco router that hosts the T1 link to the DeLeon BTS. All voice and signaling was transported over IP.

Switch (MSC) functionality for the trial was provided by a TELOS soft-switch. The first phase of the trial used a switch running at TELOS corporate headquarters in Vancouver, British Columbia, Canada. All signaling and voice traffic links between the BSC and MSC were routed over the public Internet. In late September, a TELOS switch was installed in the DeLeon central office, connected to the same gigabit Ethernet switch as the BSC and BTS. Figure 3 shows the equipment located at each site.

The Mid-Tex system was implemented in C++. The choice of C++ represents a midpoint on the spectrum of high-level languages between C (high performance and excellent portability, but weak type-checking and little support for large-scale programming), and

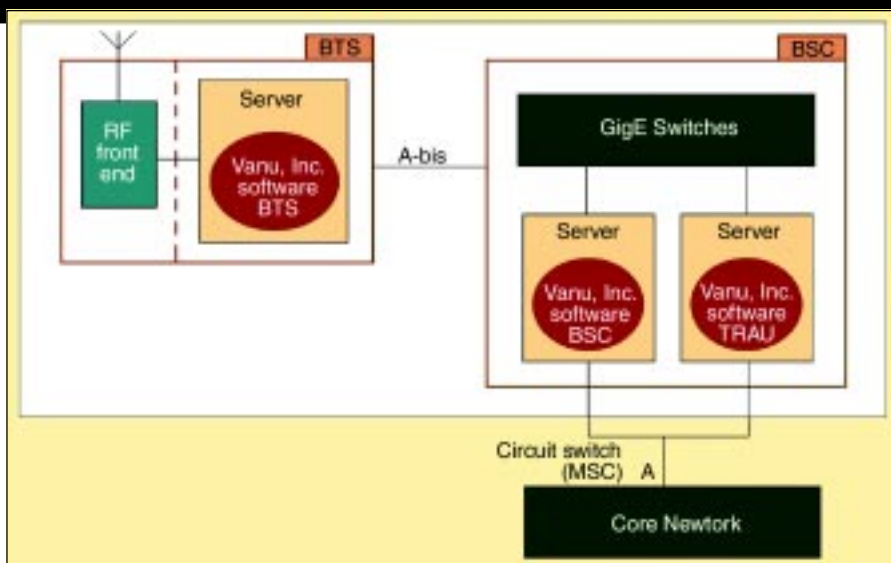


Figure 2. Base station system architecture employing software radio.

Java (modern conveniences, like strict type checking and garbage collection, but poor performance). An advantage of C++ is that the whole system, from high-performance signal processing code to network protocols to high-level control, can be implemented in one language. One drawback of C++ is exposure to certain classes of errors that are difficult to debug, such as use of uninitialized

memory, memory leaks, and improper memory accesses. On the other hand, C++ development knowledge is widespread, and techniques for using the language and addressing its deficiencies are available.

### Trial results

After the initial installation of the equipment and tests to ensure that it was function-



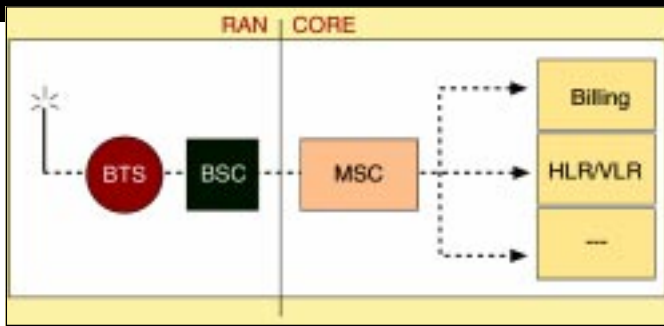


Figure 4. Cellular network architecture.

to find the balance between conserving a mobile phone's battery life and maintaining a high-quality voice link in all conditions. The first algorithm was too dynamic, frequently requesting the mobile to significantly change its transmit power every SACCH period. The algorithm frequently instructed the mobile to reduce its transmit power, resulting in poor reception by the base station. Trying to fix this resulted in a new problem: The algorithm did not increase the mobile's transmit power fast enough when the mobile was moving away from the tower at high speed. This resulted in dropped calls while driving away from the tower.

The adjustable reporting level of the logging system was instrumental in working on the power control adjustments. In normal operation, the base station does not log all of

base station on each timeslot. This allowed engineers to compare the measured power with the power control messages sent to the mobile, to determine when the power control algorithm was failing. After testing several implementations of the power control algorithm, a balance was reached that smoothed the mobile's changes in transmit power, while still increasing the mobile phone's power at a high enough rate to match the movement of the mobile traveling in a car away from the base station at high speeds.

The handover algorithm suffered from similar problems. The first version caused the mobiles to handover too quickly, resulting in many handovers back and forth between the two towers as a phone traveled in a straight-line path between the towers. In this case, a drive tester would converse with an engineer

the power measurements during a phone call, to reduce the load on the base station as well as reduce the amount of information in the logs. For these tests, however, the reporting level was increased so the base station reported the power level measured by the



Figure 5. System hardware at the DeLeon central office.

back in Cambridge, while the engineer examined the current log messages. Instructions could easily be given to the drive tester to try different test cases, and the engineer could be notified of any problems with the voice link. The handover algorithm was tuned so that handover now occurs at roughly the halfway point between the two towers.

The network architecture and physical deployment of the trial system also created a few problems for the base station system. There was not enough capacity at the Mid-Tex sites to place all of the base station system on their high availability power supplies, so several parts of the system lost power

at different times due to storms. The sudden shutdown of a BTS introduced several unseen states in the BSC, which were caught through the design-by-contract method. The contracts made the problems obvious by reporting the violations to the log. This provided a starting point in the logs, enabling engineers to determine the sequence of events that caused the problem.

Using the public Internet as the voice and signaling connection between the MSC and BSC introduced the same type of problems. The link between the MSC and BSC could sometimes disconnect for several minutes at a time, a situation that was not handled well at the start of the trial. As in the previous

case, design by contract found many of the error cases, and the messages from the logs were used to find the cause of the problem.

As engineers iterated through releases of the base station software during the three-week period, new releases were downloaded to the system over an Internet connection and could be installed in a few seconds. The ease of deploying new releases allowed the team to make many small, incremental changes with each release. Improvements were made to the BSC and BTS, across all aspects of the system. The radio interface, the voice-over-IP transport protocol, and the link detection between the BSC and MSC are examples of areas where improvements were made.

## Conclusion

The use of off-the-shelf servers and the Linux operating system also streamlined the development and deployment process. Capabilities such as log retrieval and remote software installation were built into these commodity components, meaning these did not have to be custom developed.

The deployment phase was completed in three weeks, at which point the system was ready for Mid-Tex to begin its evaluation. After extensive tests, Mid-Tex declared that it was highly satisfied with the performance of the software radio.

It offers wireless carriers significant cost-savings, while also providing an easier migration path to new standards and services. Based on the portability of the software and its ability to run on industry-standard servers, this flexible solution enables rural providers like Mid-Tex Cellular to test new revenue-generating applications with subscribers, as well as facilitate the carrier's roaming agreements with nationwide operators such as Cingular and AT&T Wireless.

While offering carriers improved flexibility and efficiency in the use of their radio spectrum, software-defined radio solutions allow carriers to upgrade to new air interface standards, and deploy new revenue-generating services by implementing software upgrades to the network infrastructure, as opposed to overlaying additional network hardware. For rural cellular operators, this is an attractive scenario that allows them to quickly and efficiently upgrade their 2G radio systems to new 3G technologies, without assuming the large capital and time-to-market costs typically associated with technology overlays.

In addition to providing rural carriers with a more cost-effective means for deploying new services, the Vanu Software Radio's ability to simultaneously enable multiple technology standards across the network also presents a compelling roaming business model for rural operators. Future wireless networks that will simultaneously support combinations of AMPS, TDMA, GSM/GPRS, EDGE, CDMA and 3G technologies will allow rural carriers to enter into roaming agreements with multiple nationwide operators, and support myriad circuit switch, packet and multi-media services. RFD

### ABOUT THE AUTHOR

Jeffrey Steinheider is the base station engineering manager at Vanu Inc. ([www.vanu.com](http://www.vanu.com)), where he develops software radio systems for cellular infrastructure. He received his Masters in engineering and Bachelor of Science in electrical engineering and computer science from M.I.T. He can be reached at [jlstein@vanu.com](mailto:jlstein@vanu.com).