

So You Want to Skim – Practical Issues in Deploying *CW Skimmer* in a Contesting Environment

As you read this, it's only been a little over 6 months since VE3NEA's *CW Skimmer* software first appeared on the market. From the strength of opinions being expressed, pro and con, you'd think it had already turned the contesting world upside down. In fact, the sum of real-world experience with *CW Skimmer* is at most a few operators, in only a few major contests.

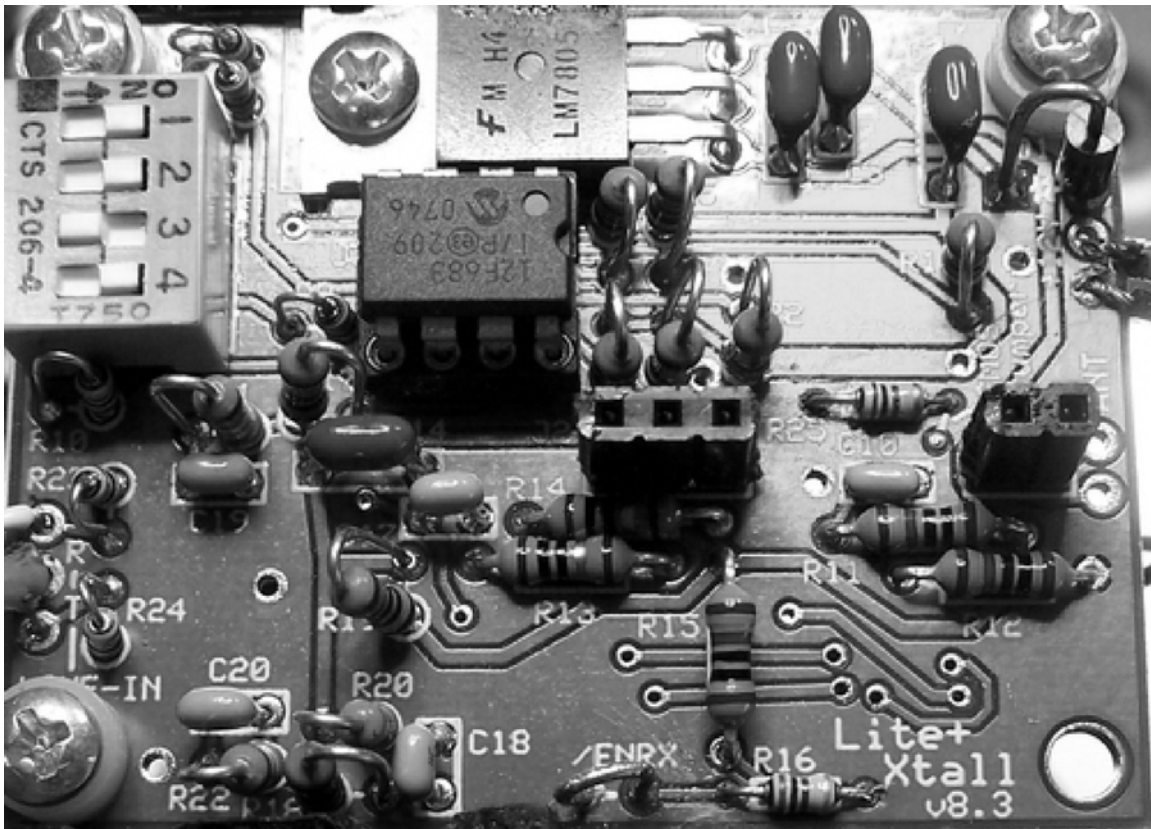
My assignment for this article was to outline the major considerations that will affect practical deployment of the software and related hardware, based on my experience and that of a few pioneers. Let's start by identifying the main questions you'll have to deal with. To me, they are:

- Choice of SDR (software-defined radio) and overall configuration
- Antenna(s)
- Computer and sound card
- Integration with contest logging software

Receivers

There are two potentially workable general solutions (more about that "potentially" in a moment). You can go with a stand-alone receiver, or you can use a single-frequency SDR as a downconverter, connected to your receiver's IF before the roofing filter.

There are a variety of SDRs available that will fill the bill as stand-alone receivers. The least expensive is the SoftRock Lite Version 8.3+Xtall, at less than \$50 in kit form (see www.softrockradio.org for details). It employs a DIP-switch-controlled frequency synthesizer and 4 plug-in bandpass filters to cover the 6 HF contest bands (fig.1).



I have been using a single-band SoftRock with *CW Skimmer* since March, and I can testify to its adequacy. To make a 6-band system that is reasonably easy to use, however, will involve some home-brewing. Single-knob bandswitching seems like a minimum requirement. In addition, you will need a sound card in your computer that has at least 96 KHz recording capability; the bandwidth determines how much of the band *CW Skimmer* can “hear” at one time.

The next step upward in the stand-alone department is the SDR-IQ receiver by RFSpace (www.RFSpace.com), which costs \$499 (fig.2)



For that you get a very competent general coverage receiver, with up to 190 KHz bandwidth from 500 Hz to 30 MHz, in a very compact package. The SDR-IQ does not need a sound card. Instead, a single USB connection handles data, control, and power for the receiver. The SDR-IQ package includes very sophisticated spectrum analyzer, called *Spectravue*. One argument for this option is the other things you can do with the unit when you're not contesting, from VLF to SWLing.

One caveat – the only documentation available for the SDR-IQ is that provided with the bundled software. There is no schematic, and operation of the hardware can only be inferred by observing what functions can be controlled by the software and what effect they have on the receiver. It truly is a “black box,” but it works very well.

Once you move further up the price scale, there is a wide range of possible hardware solutions. I won't try to run through them all, but some folks are even using full-up SDR transceivers like the Flex 5000A.

At first glance, the IF tap option looks attractive. With a single down-converting SDR, you can cover all bands. You would automatically have access to the antenna on the “tapped” transceiver, and to any bandpass filters in its receive path. The covered bandwidth would move as you tuned, potentially covering a larger “swath.”

The devil *is* in the details. To begin with, not many transceivers have an IF output on the antenna side of the roofing filter. Adding one could be a challenge, since many transceivers' first IFs are in the low-VHF region, probably requiring a mixer to get down to a usable frequency. A few radios, notably Elecraft's K3 and the Kenwood TS-940-850/950/870 series, offer such an IF output, and N8LP has

come up with a very neat single-purpose SDR, the LP-PAN (www.telepostinc.com), which is reasonable in price and performs very well without affecting the “host” receiver.

Because IF frequencies are all relative to the frequency the transceiver is tuned to, in IF mode *CW Skimmer* needs to know that frequency, in order to calculate the frequencies of the data streams it is “seeing.” That means that *Skimmer* needs to be able to connect to the transceiver’s radio control port. The rub is that normally only one program can connect to a serial port at one time. If you’re willing to forgo radio control with your contest logging program, this is not an issue, but if you aren’t, you have a problem. Some people have reported success with serial port splitter software, including a freeware offering from www.eterlogic.com. Also, MicroHam has just announced a new version of its Router software, which provides two independent virtual serial ports per radio for all of the MicroHam interfaces that do radio control – after a quick test, it seems to work FB.

Problems can also arise when the host receiver is tuned, either manually or by mouse-clicks on *Skimmer*. It can take anywhere from a few tenths to a couple of seconds for *CW Skimmer* to regain its equilibrium any time that happens, and during that interval decoding is disrupted. For this reason, principally, Alex has chosen to limit the bandwidth covered in this mode to 24 KHz, representing the maximum bandwidth that can be seen without retuning, on a VERY large monitor. As *Skimmer* evolves, this limitation may be lifted, but for now, it’s there. For this and some other reasons, I believe that the stand-alone receiver option probably deserves first consideration.

You’ll notice, when you first start up an SDR with *CW Skimmer*, that it tends to make a lot of “junk spots”, typically high in the band, up in RTTY territory. Often, you’ll notice that these “spots” replace, or are replaced by, spots for the same station at a widely different frequency. What is happening is that images result from the signals on the two audio channels (I and Q) not being exactly 90 degrees out of phase and the same level. *CW Skimmer* has a very clever automatic compensator for this. You need to you run *Skimmer* for a number of hours, with loud signals on the band, preferably during a busy contest weekend. Once it collects enough data points to construct compensation curves, the images will almost entirely disappear. Alternatively, you can use your own transmitter into a dummy load, sweeping across the band. It’s fun to watch.

Antennas

The well-equipped *Skimmer* setup will probably offer the option of switching among at least two antenna options:

- A dedicated omni-directional receiving antenna, such as an all-band vertical. Some multi-multis will probably use the same beverages or other

specialized receive antennas they now use for in-band S&P. You will need to protect against transmitted RF, at least enough to ensure that the SDR isn't damaged. A simple protector using back-to-back diode strings and/or a #47 light bulb should be adequate(http://www.pvrc.org/~n4zr/Articles/simple_protection_for_the_fl_edgl.htm). RX bandpass filters are another possibility, with obvious additional benefits.

- Sharing the RX antenna of your transceiver – either the main one in an SO1R station or the radio used principally for S&P (in SO2R). This solution has the advantage of using whatever stubs or bandpass filters you use to protect your second radio, and the disadvantage that the “host” radio must be on and switched to the band you want to skim. You'll need to experiment to make sure that the SDR doesn't desensitize the “host” radio – the dreaded “suck-out.” If that occurs, the simplest solution may be to use one of Clifton Laboratories' Z-10000U buffer amplifiers (www.cliftonlaboratories.com), between the receiving antenna loop and the SDR. This will also silence the local oscillator of your SDR (if it has one) – mine went from S9+35 on the “host radio” to inaudible. Another approach is to use a passive splitter such as those made by Mini-Circuits in the receive antenna loop. This will reduce the signal to both your receiver and the SDR by 3+ dB, and doesn't isolate the SDR's local oscillator, but it is simple and pretty bullet-proof.

Computer and Sound Card

My minimal 2.4 GHz Dell computer's horsepower is somewhere close to the minimum necessary to run *Skimmer* on a busy contest weekend with a lot of signals to decode. I ran it during WPX with the number of decoders limited to only 150, and frequently saw CPU utilization in the 75-80 percent range. I use a Winkey to generate CW; otherwise, I suspect there might have been problems with CW quality.

One solution is simply to dedicate a computer to *Skimmer*, and use a network to pass spots to your logging software. Computers like my Dell are probably available for only \$200 or so. Another approach is to get a computer with a dual-core CPU. These are available now for very reasonable prices, and will permit you to assign *Skimmer* to one core and your logging program to another, which should finesse any CPU loading problems. Some users have mentioned that video capability could be a problem – Alex advises minimizing *Skimmer* when not actively using the waterfall, and keeping it as narrow as possible when doing so.

To test the capability of your current computer, Alex suggests running his 2005 CQWW sample recording – to do so does not take a sound card or anything

except a trial copy of *Skimmer*. If you leave the decoder function on “adaptive” and you do not see at least 500 decoders in one run through the recording, then your computer is marginal – mine topped out at 450 on this test.

If your receiver setup requires a sound card, E-MU (Creative) makes an external USB sound card (the 0202) with 192 KHz recording and playback that looks pretty good for around \$100. M-Audio made some decent PCI cards (now discontinued) with 96 KHz recording capability for as little as \$50, and you can probably still find these on E-Bay for even less.

Integration with Contest Logging Software

CW *Skimmer* incorporates a basic Telnet server. You can log onto it locally, just like another DX cluster, and feed spots to your logger. All of the major Windows contest logging software packages accept spots as formatted by *Skimmer*. If you want to feed spots from more than one *Skimmer* at once, or from *Skimmer* and telnet simultaneously, you need to merge two or more data streams. K1TTT's WintelnetX freeware (<http://www.k1ttt.net/software.html>) will do this job for you.

At present, *Writelog* is the only major contest logger that fully integrates with *Skimmer*, due to the foresight of the *Writelog* programmer and the efforts of Ed, W2RF. You can click on the *Skimmer* bandmap and control a radio through Writelog, and Writelog can report the status of a given call (dupe, mult, etc.) on the *Skimmer* bandmap. Pretty neat - but it's not at all clear if or when other major Windows loggers will implement similar capabilities.

Even if you use a standalone SDR with *Skimmer*, you will need a means of controlling its frequency and telling *Skimmer* what frequency it is on. What I do is to manually switch my SDR to the proper band and then use a super-minimized *Skimmer* window on-screen with my logging software, to tell *Skimmer* which band I'm on.

Operating

I'm not going to go very deeply into operating, because we're all still learning how best to make use of CW *Skimmer*'s capabilities. We'll know a lot more by next spring. Any operator is well advised to take advantage of the “cumulative” way *Skimmer* operates. When you transmit, *Skimmer* will probably just stop working – at least it won't spot you to your own bandmap, because it has built-in protection against self-spotting via the Telnet server. But it only takes a minute or two for *Skimmer* to capture most of the calls on a busy band, and any time you give it a chance to listen, it will find new spots or update old ones.

For the SO2R operator, it may be best to use *Skimmer* like packet, populating the bandmap on other open bands while you run on your primary band. It stands to reason that you don't want to become a *Skimmer* operator - it could raise the concept of "single-op distracted" to a whole new level. Let it work in the background as much as you can.

For the SO1R operator, and perhaps for all S&P operations, Alex advocates using *Skimmer* differently. He says,

The true purpose of the *Skimmer* is to help the skilled hams who do S&P manually to do this more efficiently. The most important part of the program is the "annotated waterfall display" - that is, a high resolution waterfall with the callsign labels attached to the signal traces. The operator is supposed to use this display to find the signals on the band, instantly tune at a station with a mouse click, then copy the signals by ear and work the station. The labels are there just to help the operator quickly decide which stations should be copied first, and which ones can be safely skipped.

Ideally, if you want to do this you should solve the "one-radio – two controlling programs" problem, but a near-term option is simply to forgo computer control from your logging program, and rely on *Skimmer's* bandmap to get the job done.

I'm sure that during the coming contest season plenty of folks will be experimenting with *CW Skimmer* and improving on what I've suggested above. Let me know what you come up with, and I'll put together a follow-up article for next summer.

Thanks to VE3NEA, K8ZOA, K4TD and DL6MHW for their input and comments on this article. Any remaining mistakes are all mine.