



DCC Impulses Column

by Bruce Petrarca

Photos and illustrations as credited

An interview with some of the folks behind JMRI.

This month's column deals with a corner of the DCC world: JMRI. While folks know a lot about the DecoderPro suite from this extensive software package, there is a lot more software behind the curtain (<http://jmri.sourceforge.net/help/en/html/apps/index.shtml>). Also, there is an army of volunteers who "make it happen."

Also, there are lots of people (<http://jmri.sourceforge.net/help/en/Acknowledgements.shtml>) and their stories there, too. I conducted interviews with a few of the hundreds of folks who work to bring JMRI to you as a free-to-the-user application. However, I recommend that everybody who uses it contribute, say \$10 per year, to the cause (<http://www.decoderpro.com/donations.shtml>). You will see why later in the column. BTW, I'm contributing my income from this column to JMRI.

First, we start with Bob Jacobsen. Bob is the focal point for JMRI. He coordinates the releases of the JMRI package and has tirelessly worked JMRI and the concept of open-source software. I've communicated with him via eMail for years. It was nice to have dinner with him at the 2011 convention in Sacramento.

Bob Jacobsen discusses JMRI history

[1] Bob Jacobsen - photo courtesy Bob Jacobsen



MRH - What does JMRI stand for?

Jacobsen - It doesn't really stand for anything anymore, it's become just a name. Originally, it was intended to be a library for programmers to use, in the Java language, so it got the name "Java Model Railroad Interface", JMRI for short. But now it's much more aimed at individual users with things like DecoderPro and PanelPro. There are a few programmers using it as a library. CATS is the best known result of that, but there are a few people using it with their own code for their layouts.

MRH - Explain, please, about Java and how to run JMRI.

Jacobsen - The "Java Runtime Environment" (JRE) is something that's needed for a computer to run programs written in the Java language. Some computers come with one, some have to have it installed. On Windows, the JMRI installer will look at what's already present and prompt you through what needs to be done. Linux machines usually have a suitable Java installed, but if not you can install it with the tools built into your Linux distribution. On Mac OS X, Java has been installed automatically for a long time, but this is starting to change. That's going to be a confusing situation for a little while.

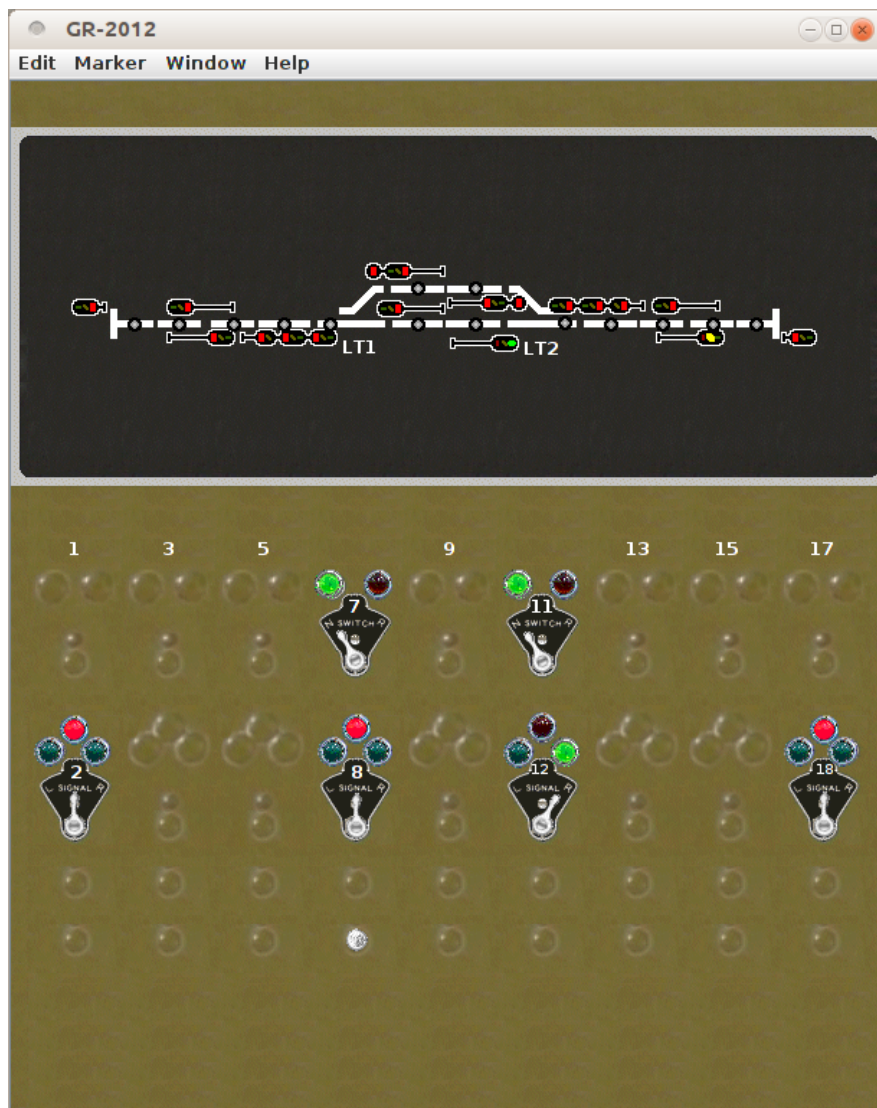
Petrarca - I just installed JMRI on a couple of Macs with Mavericks (OS X 10.9) and it was pretty easy. I checked for Java, using their site (<https://www.java.com/en/download/installed.jsp>) and installed it where needed. Then I downloaded the JMRI

file from the web site (<http://jmri.sourceforge.net>) and installed it. No other fiddling required.

MRH - What are the major sections of JMRI and who is the lead team member for each section?

Jacobsen - We don't really have "lead team members". We have people who like to work on things. Some parts of JMRI were constructed by lots of people; more than 40 people have done parts of the decoder definitions in DecoderPro. At the same time, there can be a few people who've done a lot of work in an area. Michael Mosher, for example, has done a large part of the decoder definitions. As another example, Kevin Dickerson and Dick Bronson have done a lot of work on the prototypical "Signal Mast" support, but a lot of other people have contributed too. David Duchamp started Logix and continues to work on it, as does Pete Cressman. Dan Boudreau has done most of the work on operations.

[2] PanelPro emulates a classic CTC Panel - photo courtesy Dick Bronson



The code that makes up JMRI is common property, in the sense that anybody can contribute a change to any part of it, and developers can improve any part. The idea is to encourage people to improve it, not build structures that make it harder to improve the code. Whether it's a patch contributed by somebody new, a little improvement that somebody thought of one afternoon, or a big project, JMRI needs continuing improvements to keep becoming even more excellent.

MRH - Tell us a little about JMRI's history.

Jacobsen - JMRI emerged from a couple of independent efforts around 2000-2001. Technology had reached the point where it had become possible for a small group of people to get together online and develop software even if they weren't in one place. By pulling together some work on decoder programming (the original DecoderPro), interfacing to DCC systems (the original JMRI) and some other bits, the early versions of JMRI could do something for a wide range of people. That in turn attracted more people to add little bits, which got more people interested, etc. Within a couple years, it was a nice little community working on something of value to hobbyists.

After DecoderPro, the next big thing to be added to JMRI was control panels. The original idea was to just have a few icons on the screen that would let you do simple things on the layout, but through the work of Dick Bronson and others it rapidly grew to allow emulation of a complete US CTC machine. That, along with the signaling support that was being developed in parallel, then laid the groundwork for the "prototypical signaling" development work that's going on now. This is now known as PanelPro.

More recently, JMRI has continued to grow through new efforts that certainly weren't imagined back at beginning such as wireless throttles and the operations support. In part that's because all the code is available to everybody, so you can build new things on top of it pretty easily. In part it's because general technology, like iPhones and WiFi are becoming more common and more powerful. But the most important thing is that JMRI is still a nice place to work on things like this. That attracts new people, and makes longer-term members more enthused because it's still fun to work on.

MRH - Where to I buy JMRI?

Jacobsen - You don't have to buy JMRI. You already explained how you went about installing it on some Macs. The same method works for Windows or Linux.

The really adventurous can also download a copy of the source code from the JMRI web site and build their own version.

If you don't want to download it, you can get a CD or USB stick with JMRI on it from several places. Your former business, Litchfield Station, was one of the first to offer this service to its customers. These people tend to offer them for a very small price and don't make any money on them, so they generally prefer that you download it if you can. It's just easier for everybody.

MRH - How does one become a member of the team?

Jacobsen - Do stuff. Seriously, that's all there is. Help answer user questions, write a better explanation for the help pages, improve a decoder definition, or write some code.

MRH - You were part of a landmark legal settlement, involving software rights. Would you explain, please?

Jacobsen - Starting in 2004, somebody wanted to be paid for every copy of JMRI downloaded, because he claimed he had patents that cover some of the things JMRI (and other programs) do when talking to DCC systems.

This couldn't be allowed to happen, because it would have resulted in the destruction of the JMRI community and the JMRI program. So, I fought it in federal court with the help of Victoria Hall and David McGowan, two fantastic lawyers, and the team they assembled.

Two illegally obtained patents were eventually invalidated at the Patent Office, but we also had to appeal to the World Intellectual Property Organization to get the decoderpro.com domain name back, and file copyright actions over misuse of the DecoderPro decoder definitions.

JMRI is available "free", in the sense of no-cost, but not "free" in the sense of "no restrictions"; there are a few restrictions on what you can do with it in the license. One of the things you can't do is claim it as your own work. A lot of people have worked on it, and they deserve to be recognized for that. If we were to allow somebody else to claim it as they own, a lot of the joy of working on JMRI would go away, and the community would be badly damaged.

These aspects of Open Source law had never really been tested in the courts, and we ended up having to take the case to the Court of Appeals for the Federal Circuit in Washington DC. That was a really time-consuming and nerve-wracking thing, but it had to be done, and in the end we won. A settlement agreement covered the last parts, and it was finally over in 2010.

Folks wanting more details can find them on the web (<http://jmri.sourceforge.net/k/summary.shtml>).

MRH - What were the financial impacts of this lawsuit and upon whom?

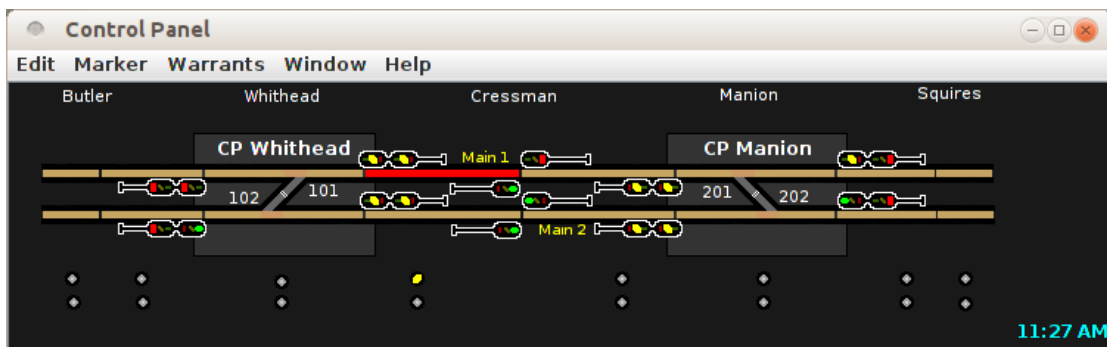
Jacobsen - The costs of a federal lawsuit are pretty severe. The settlement dollars helped, but I'm still significantly in the red.

MRH - Since this benefitted the group, how can JMRI users help?

Jacobsen - Contribute. There's a contributions link on the JMRI web page. Contributions go first to cover the costs of running JMRI (web site & domain name fees, copyright registration fees, etc).

Petrarca - The remainder then goes to reimburse you for your 5-digit legal expenses. This decision was very far reaching in the area of open source software. Not only will your contributions help JMRI, but there are lots of bits of "free" software that will benefit, at least obliquely. I recommend folks give a small amount, say \$10 per year, if they are using any part of JMRI.

[3] PanelPro emulates a modern ABS panel - photo courtesy Dick Bronson



MRH - How do I use JMRI with my DCC system and what DCC systems can interface with JMRI?

Jacobsen - Most DCC systems that have a computer interface of some kind can be used with JMRI. There's a list here: <http://jmri.org/help/en/html/hardware/index.shtml>

The manufacturers of the most common ones are uniformly helpful people who've worked with JMRI developers so that their systems are well-supported. Look up the page for your particular equipment to see what kind of interface you might need, if there are any limitations, etc.

For programming decoders, DecoderPro basically works like a throttle on your DCC system. So if you can program the locomotive manually, DecoderPro will be able to take over for you and automate that.

MRH - What about MRC systems?

Jacobsen - MRC had gone its own way, relying on its own software and not providing the code so that JMRI can interface with their system. However, in the spring of 2014, they decided to share the interface specifications with the JMRI group. So the MRC system support will begin with the 3.9.1 test release version of JMRI. That or any later version will do the trick.

MRH - Many readers are concerned about being left out in the cold starting out with a new computer program. What help is available to them?

Jacobsen - The JMRI community is large and helpful. A good place to start is either at a local club, where you'll likely find somebody who can help you in person, or by joining the JMRI users mailing list (<mailto:jmriusers-subscribe@yahoogroups.com>) and asking there. Whether the question is simple or complicated, there's somebody around who'd like to help you get it answered.

MRH - Thank you, Bob for that tour.

Now, let's talk with Michael Mosher, who, in addition to working on decoder definitions has a lot of information about using JMRI as a wireless throttle network. Although we've exchanged eMails, I have not had the pleasure of a face-to-face meeting with him.

Michael Mosher on using a mobile device as a throttle

[4] Michael Mosher - photo courtesy Michael Mosher

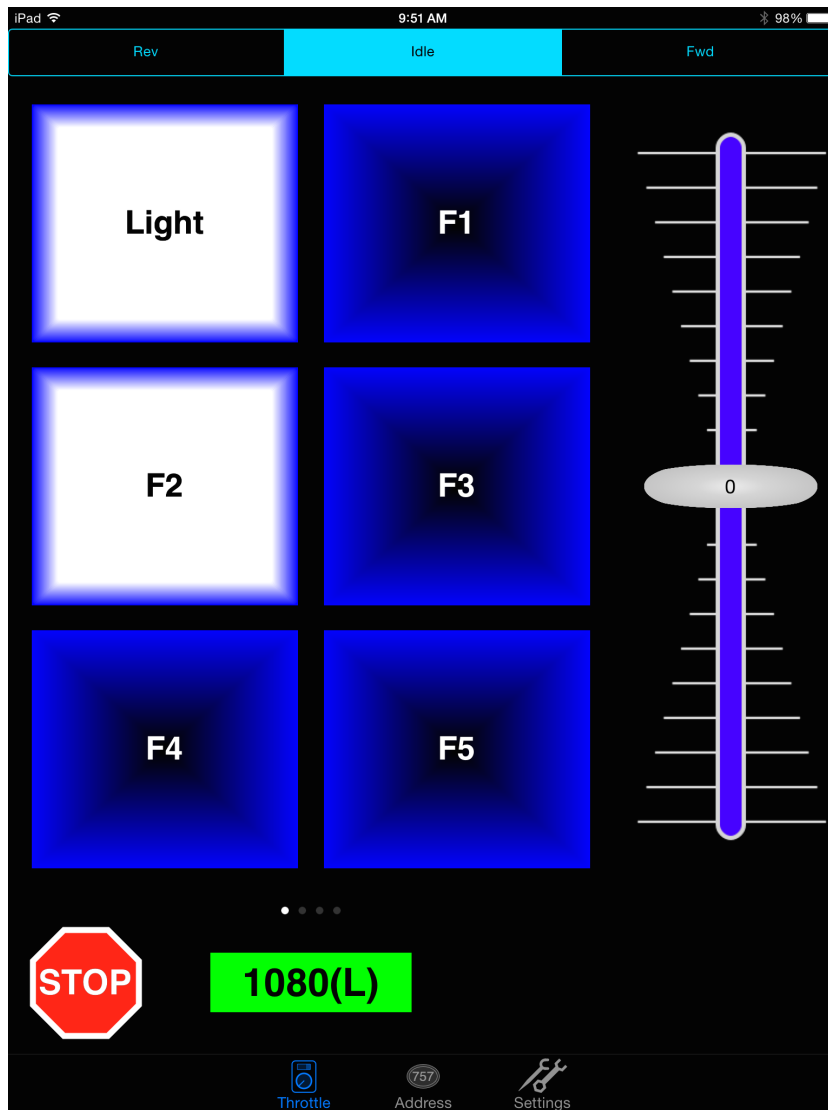


MRH - Tell me about how I can run trains with my cell phone or tablet. What hardware is needed to take advantage of this, assuming you are starting with JMRI connected to your DCC system?

Mosher - You can use JMRI as a portal between a WiFi network and your DCC system, allowing you to run trains from your portable device.

If your computer is connected to a WiFi network, then make sure your cell phone or tablet is connected to the same WiFi network. Open the app on your phone and you should see the layout advertised.

[a] WiThrottle Lite running train 1080 with Yard Mode enabled - capture by Bruce Petrarca



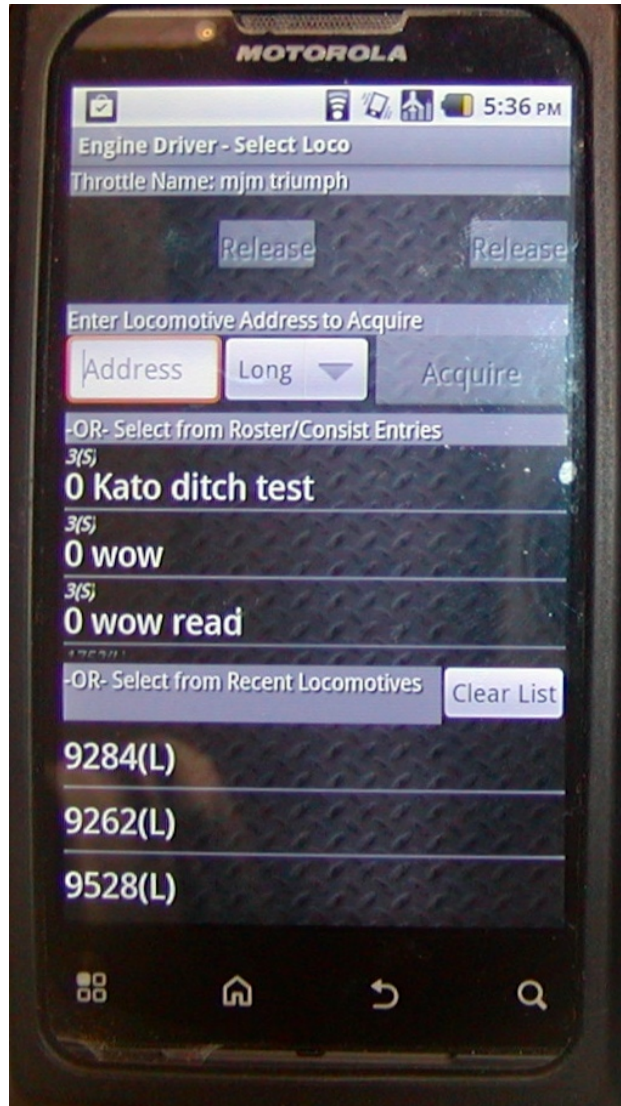
If the computer is not already on a WiFi network, then a WiFi access point will be needed, typically a WiFi router. Internet connection is not required for throttle operation. The JMRI computer can be connected to the router either over WiFi or via a wired (USB or Ethernet) connection.

If you have a smart phone already, you can use it as soon as you download the appropriate app. See the sidebar for app information and inexpensive sources for usable devices.

Start the WiThrottle Server by clicking the menu Tools - Throttles - Start WiThrottle or the menu Debug - Start WiThrottle.

Once in the app, give it a moment for auto-configure to find the WiThrottle server then select it. There will be a list of locos based on DecoderPro roster IDs that you can select from or enter any address. If the Function Labels tab has been filled out for the selected loco then the function buttons will have that text rather than just a number.

[6] Engine Driver running on an Android phone - photo courtesy Michael Mosher



I've also included some data on the frequency allocations in the 2.4 GHz band. Check out the sidebar, if you are interested.

MRH - Thanks, Michael. Now we know another use for our phones.

Sidebars for Michael's comments

Portable devices and apps

For throttle devices, they will need to be connected to internet long enough to download the app. Once the app is installed, no internet connection is needed.

For iPhone, iPod Touch, iPad, go to App Store and search for WiThrottle Lite for free basic functions app or WiThrottle [a] for paid full feature app.

For Android version 1.6 or later go to Google Play Store and search for Engine Driver for free full feature app. If Android tablet does not have access to Play Store or is version 1.5 or earlier, then download Engine Driver app directly from <http://enginedriver.rrclubs.org/>

The cell radio can be turned off by setting to airplane mode, battery life will be better with it off. WiFi needs to be turned on and connected to the same WiFi network as the host computer.

Since the cell radio is not needed, a source for low cost throttles is to look for used phones that have bad ESN or otherwise disabled or non-functioning cell radio.

2.4 GHz spectrum allocations

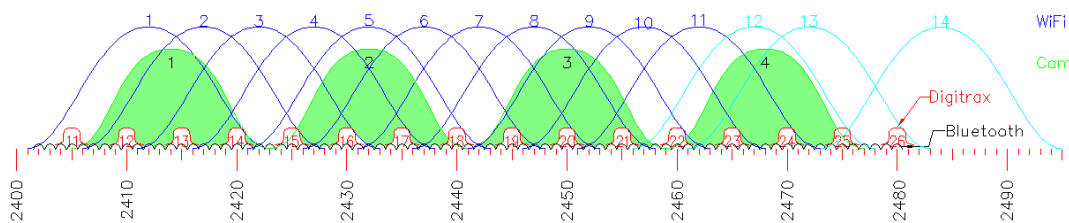
The non-licensed portion of the 2.4 GHz spectrum is very crowded with lots of overlapping services. In order to avoid interference, you need to carefully choose the frequencies you use.

In the US, WiFi has 11 channels, as shown in dark blue. Channels 12, 13, 14 (light blue) are used outside the US. WiFi channels are overlapping, so you cannot use, for example, channels 1 & 2 in the same place at the same time, as they interfere. The largest group of non-overlapping channels is 1, 6, & 11 (and 14 if in a country that allows that channel), therefore those are the most used channels.

Some wireless video cameras frequently used for train cams are also in 2.4 GHz band in four channels, as shown in green. The cameras in the 1.2 GHz and 5.8 GHz bands will not interfere.

Digitrax duplex radio throttles also use the 2.4GHz band in 16 channels (11-26 - shown in red). Select channel 26 if possible, as that won't conflict with any US WiFi channel or camera channel.

Bluetooth is also in the 2.4 GHz band but is a frequency hopping protocol so it will find a free space for itself.



Dick Bronson is a manufacturer of support electronics for model railroads through his company RR-CirKits, Inc. (<http://www.rr-cirkits.com>) and a MRH sponsor. Linda and I met Dick and his wife, Karen, at one of the NMRA national conventions, perhaps Seattle in 2004. We have enjoyed time together whenever we have attended the same convention ever since.

Dick Bronson discussing Prototypical signaling

[7] Dick Bronson - Photo courtesy of Dick Bronson



MRH - How did you get involved with supporting JMRI.

Bronson - My first contact with JMRI signaling was a clinic that Nick Kulp gave at Timonium. During the clinic he not only managed to completely delete the panel that he was demonstrating, but also to recover well enough to show signals changing on his demo layout.

We had just recently started our company with some products aimed at detection and signaling, so I was very interested in this new 'Open Source' RR software concept. I was already familiar with some other open source software from my day job, (Apache and Linux) so I knew the concept could work to create useful results.

Essentially the idea is that a group of people work as a team to build something that they all want to have and use, but don't have the individual time and/or ability to create on their own.

As a fledgling company we decided that what we needed from the growing JMRI community was contact with potential new customers for our new products. In exchange what we could contribute to the project was training for new users, like Nick had done for me. Because our new products supported signals we headed off in that direction.

The PanelPro section of JMRI has grown out of this cooperation.

MRH - Have you done other things than give clinics for the JMRI project?

Bronson - Actually yes. The first time I gave a JMRI clinic on signaling the program didn't work like I expected it to at all. It had the basics of what we now call SSL (Simple Signal Logic) but there were some common signaling situations that did not have any way to set them up. I was able to dig around in the code enough to add in the missing options. Some one else (I suspect probably Bob Jacobsen) took what I had cobbled together and straightened it out into an acceptable format. This emphasized to me that JMRI really is an open project that honors the efforts of anybody that is willing to make a contribution.

[8] SP's Canyon Dispatch machine in the day - photo courtesy Bob Jacobsen



As my efforts to present clinics grew we found that there was an opportunity to improve on and create new graphics for the creation of CTC panels. This was back in the days when the only way to create a JMRI CTC panel was to paste together multiple images of the background, track, levers, and lamps.

MRH - How has this 'open source' software concept worked out for you as a manufacturer?

Bronson - The synergy between RR-CirKits and JMRI has actually been critical to our growth as a company. I would like to think that this give and take has worked to improve JMRI as well.

One specific area for us is in the JMRI signaling support. Originally the only JMRI support was for direct drive signal lamps (CMRI) and for LocoNet (Digitrax) signal heads. One of our LocoNet products (the RR-CirKits LNCP) optionally included support for signal masts using the NMRA accessory signal decoder commands, but there was no way to actually use it.

One of the JMRI programmers, Kevin Dickerson from the UK, was trying to add the concept of controlling signal masts as a basic unit to JMRI. It was a natural thing for Kevin to add in the NMRA DCC signal accessory decoder support found in the LNCP into his new signal mast concept. That support has in turn led to other manufacturers designing signal mast hardware using the NMRA signal accessory decoder definitions.

This DCC signal aspect definition has been part of the NMRA DCC specifications for many years now, but has suffered the classic 'chicken or egg' syndrome. Why build a product with no way to control it? Why spend the time to figure out how to control something that doesn't exist. Alex Shepherd and Kevin Dickerson, both JMRI developers, broke the deadlock and I expect that soon the modeler will see a new class of signal decoder available from several different sources.

I just took a look at the current list of JMRI contributors (<http://www.jmri.org/help/en/Acknowledgements.shtml>) and counted representatives that I recognize from over half a dozen different manufacturers. Apparently we are not the only company that has been helped along by the existence of JMRI.

Some commercial software has existed for controlling a modeler's hardware before JMRI came along, but the relatively high cost of the commercial products greatly limits the size of their user base. As a small business we could never have grown, or even survived, with the number of customers that could afford to purchase commercial software to run our products.

[9] This model of the Canyon Dispatch machine [8] is being operated by Otis McGee on



his layout. It works through JMRI's PanelPro to control signals and turnouts - photo courtesy Bob Jacobsen

MRH - Why would the modeler want signals in the first place, especially prototypical ones?

Bronson - Before the advent of command control systems, especially DCC systems, operating signals of any kind were the exception on model railroads. Most folks were satisfied with sticking a few dummy heads at appropriate locations to complete the scenery. Because flashing lights are interesting, we sometimes even animated them to some degree. They were not an important part of operating our railroads because running one train into another was not a big problem when you only had one or two power packs to work with, and trains always went the same way at the same speed on the same tracks.

Now in the last decade or so we have moved much closer to the prototype in our train operations. Our controllers now only control a single train, and it has no limits on speed or direction when compared with any other trains on the same piece of track. It has become much more important to operate our trains like the real railroads do. There is a growing interest in 'operating sessions' where folks run their trains like a real railroad does. That of course includes not running into other trains. Suddenly it is

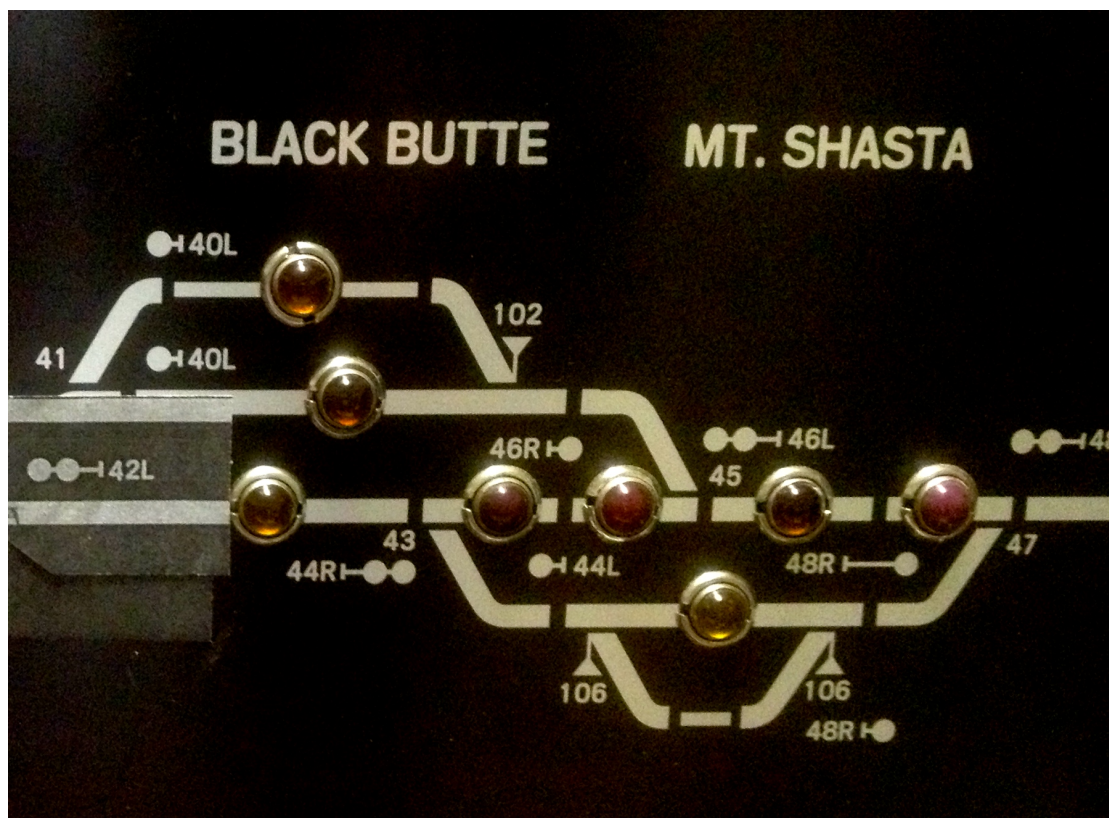
important to figure out how the real railroads managed that, and one way was to use signals. It is no longer just a nice addition to the scenery, it has become important to the safe operation of our trains.

Simple signals may be all that we actually require as modelers, but for many folks, if you are going to add signals, then you may as well do it correctly. Few would think of sticking a coal tender behind an SP cab forward, nor would they put searchlight signals beside the tracks on a Pennsy layout. The latest JMRI versions recognize this and are starting to include built in rules for the prototype signals of different railroads and eras. Like everything else in JMRI, once someone does the work required to codify the rules for a particular signal system, they become available to any one else that wants to take advantage of them.

MRH - Just how prototypical can you get with JMRI signaling?

Bronson - About as prototypical as you may ever want or need.

[10] Close up of the Black Butte / Mt. Shasta sections of the machine in [9] - photo courtesy Bob Jacobsen



I happen to know that one current Class I railroad has just started using JMRI to control the signals on their latest training layout for engineers and conductors. In other words some real world railroaders are now learning how to follow signals controlled by JMRI before they are allowed behind a throttle on the main line. Not only that, but their officials were simply amazed to discover that such sophisticated software is available to any hobbyist at the cost of a free download. This railroad uses

the NORAC signal rules, so we are talking about signals capable of displaying over a dozen different aspects, not just simply stop and proceed.

One of the instructors told me that it is not uncommon to see the trainees literally break out in a sweat as they try to remember the correct radio traffic while they make up a train or what speed to approach a signal and what reports to give as they do. They forget that it is actually just a model train they are running. For them it is a very serious undertaking to learn how to do their jobs efficiently and safely.

MRH - Thanks, Dick for your insight.

There you have a lot of the back story of JMRI. Perhaps more than you ever wanted.

Please join the conversation after the column on the MRH web site. A lively discussion frequently ensues. If you liked the column, please rate it **awesome**. You can do both with the reader feedback links at the beginning and end of the column.

Until next month, I wish you green boards.