

Intro to Decoder Programming with JMRI/DecoderPro

Dave Duchamp

Bob Jacobsen (Co-Presenter)

Dick Bronson (Co-Presenter)



What is JMRI?

JMRI (Java Model Railroading Interface) is open source (no cost) software for connecting a model railroad layout to a computer, and performing various model railroading tasks via the computer.

JMRI was/is developed by a group of volunteer programmers under the leadership of Bob Jacobsen.

JMRI uses the Java programming language.

JMRI continues to grow . . .



How is JMRI Organized?

JMRI has of an extensive library of model railroading software, and several front-end applications focusing on different areas of model railroading.

All JMRI applications use this common library.

JMRI Applications include:

DecoderPro - Programming DCC decoders.

PanelPro - Layout display for running trains.

Other applications (JmriDemo, LocoTools, etc.)



What Computer Systems are Supported by JMRI?

Windows - Vista, XP, 2000, 98SE

Macintosh - MacOS X

Linux - several flavors



What Model Railroading Tasks are Supported?

Programming DCC decoders

Computer Panel Displays (including full CTC Panel)

Computer throttles

Consisting

Control of Turnouts (Including Optional Feedback)

Routes (Controlling groups of Turnouts and/or Sensors)

Logix (Control and Automation Logic)

Control of Layout Lighting

Fast Clock

Control of Signals

and More ...



How do I get started?

1. Connect your computer to your model railroad layout.
2. Download JMRI.
One download contains all JMRI applications.
Large download - CD's are available.
3. Configure JMRI.
4. Test communications.

Detailed instructions for various computers and model railroading systems are on JMRI web site.

No computer programming is required.



Computer Connection Example

Workshop system:

Digitrax DCS100

Locobuffer II

Serial to USB adapter

Macintosh PowerBook G4



Configuration Panel

Preferences...

Layout connection: LocoNet LocoBuffer-II

Serial port: (None)

Baud rate: 19,200 baud (Sw1 off, Sw3 off)

LocoBuffer-II connection uses hardware flow control (recommended)

Command station type: DB150 (Empire Builder)

GUI style:

CDE/Motif Metal Mac OS X

Programmer defaults:

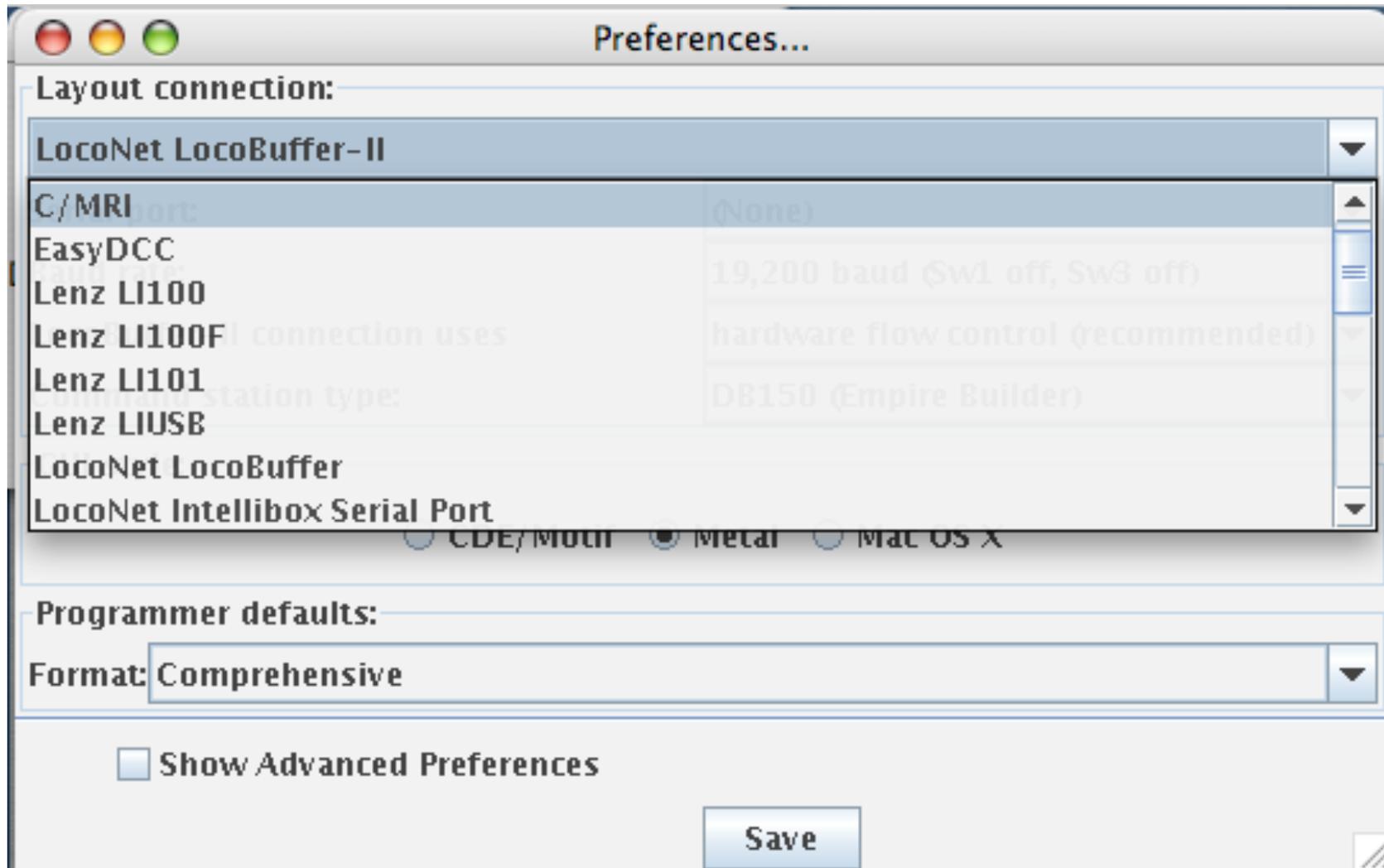
Format: Comprehensive

Show Advanced Preferences

Save

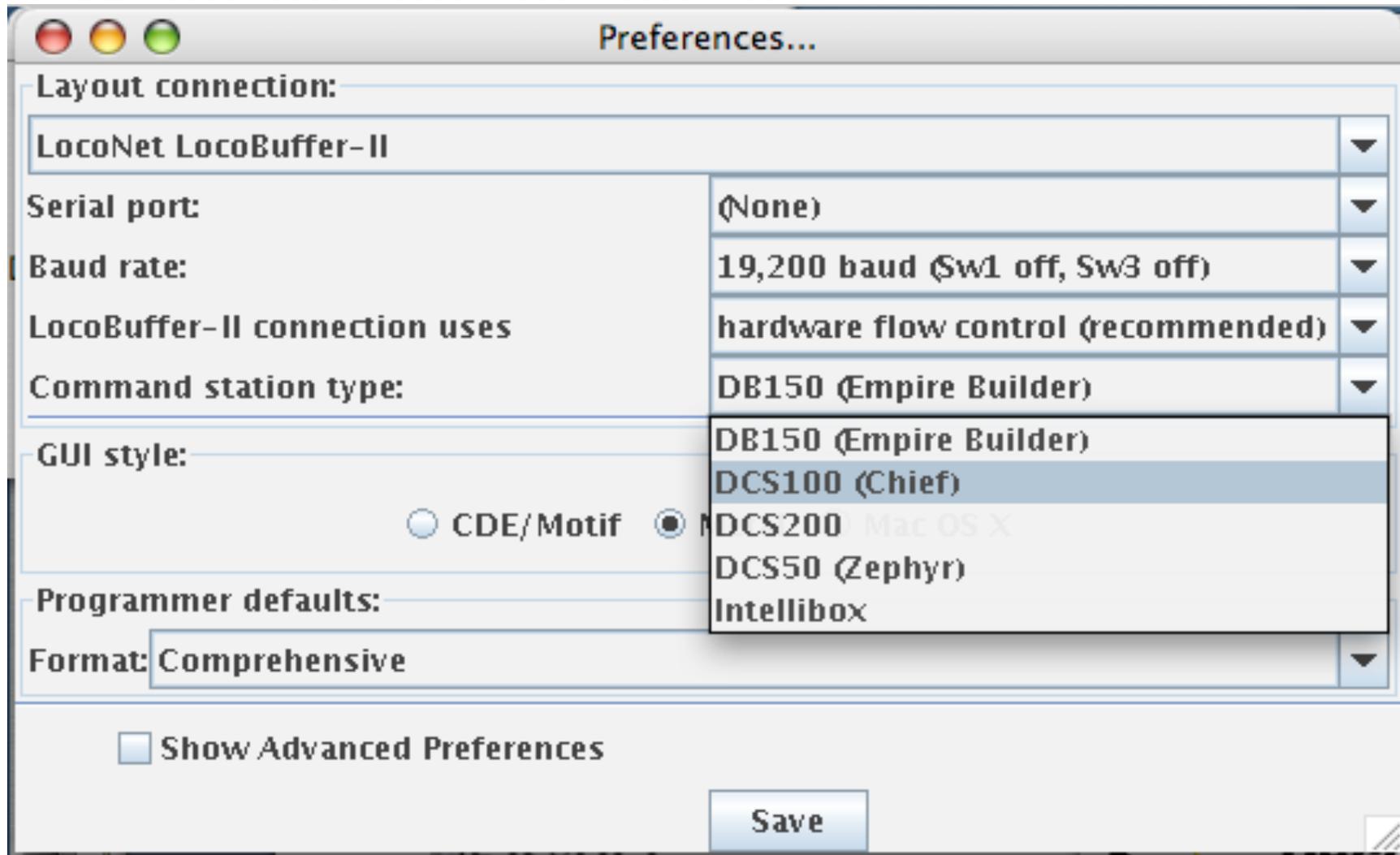


Select the type of layout connection from an extensive pull-down menu.



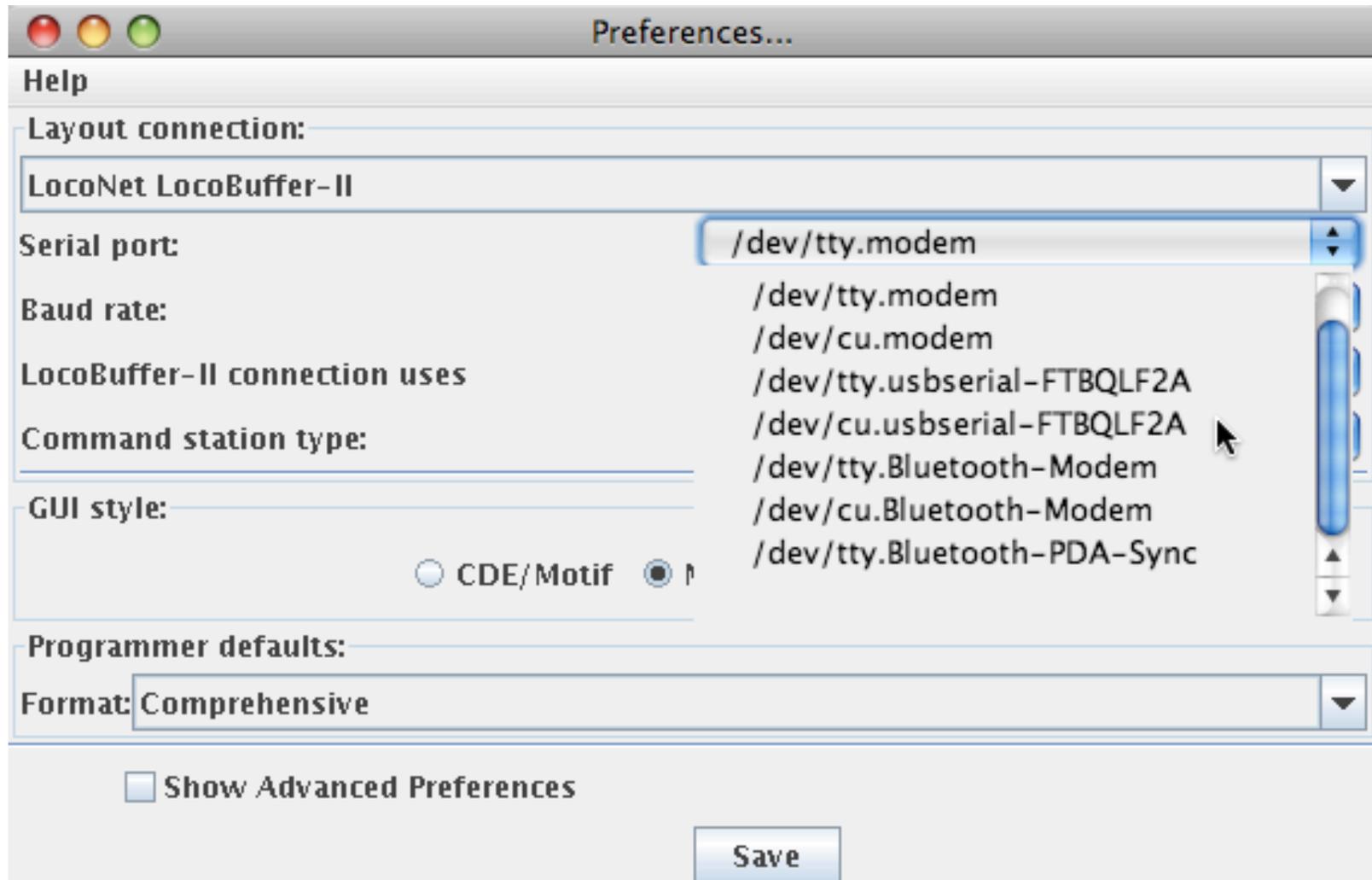


Select the command station type from the menu of types compatible with the layout connection.

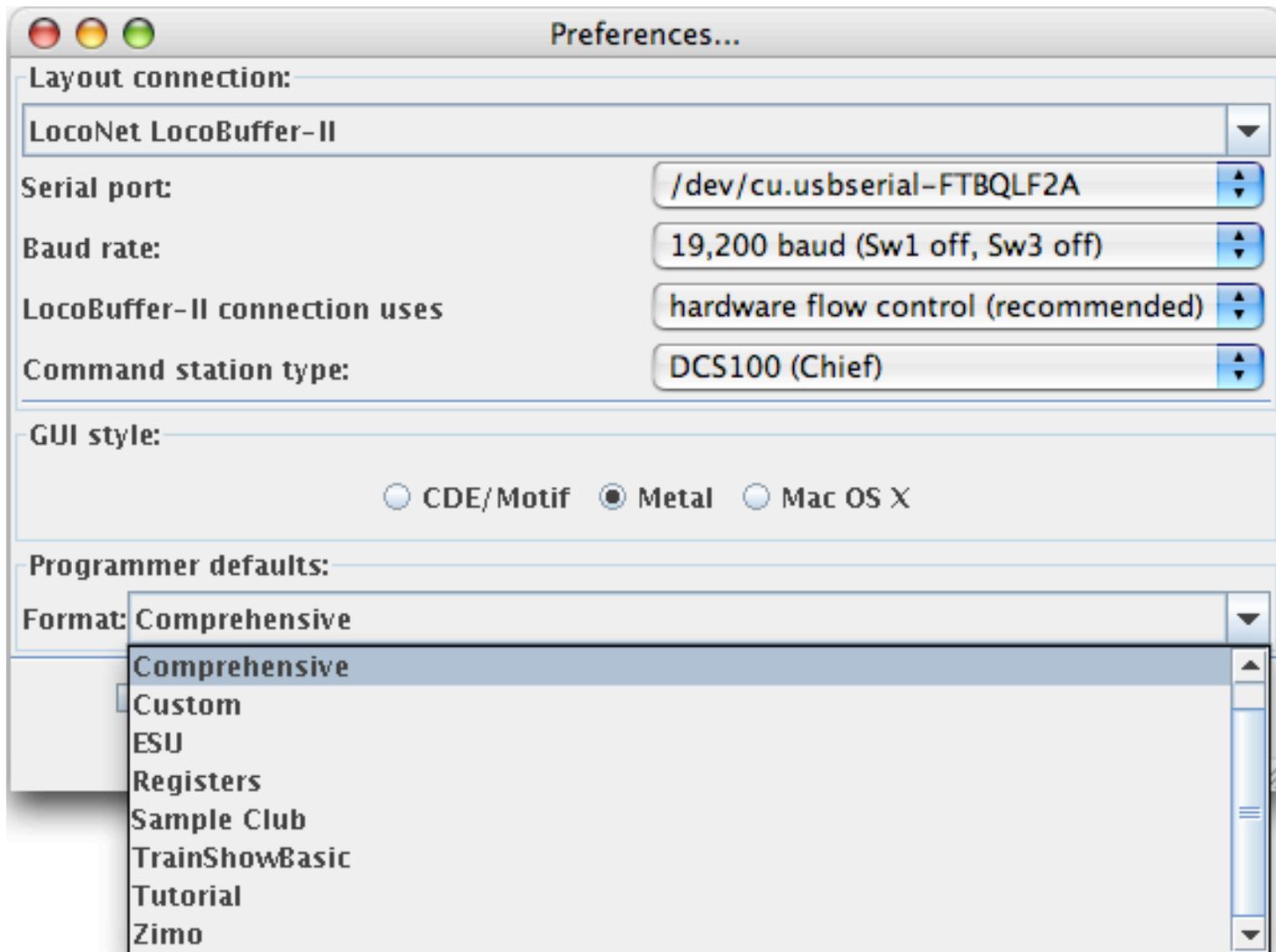




Select a serial port from the menu listing the serial ports that JMRI can detect.

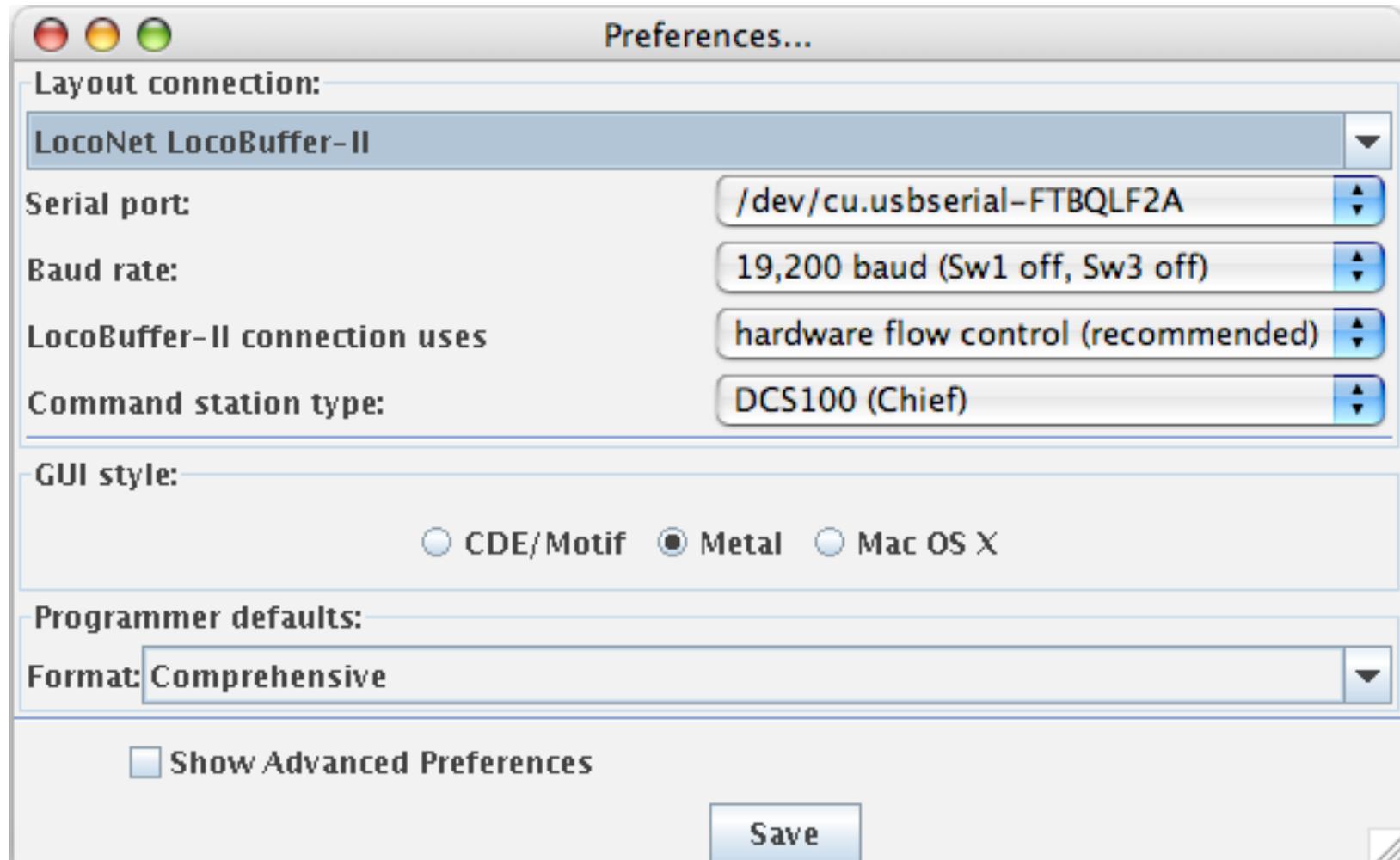


Select a default programmer (Comprehensive is usually best).



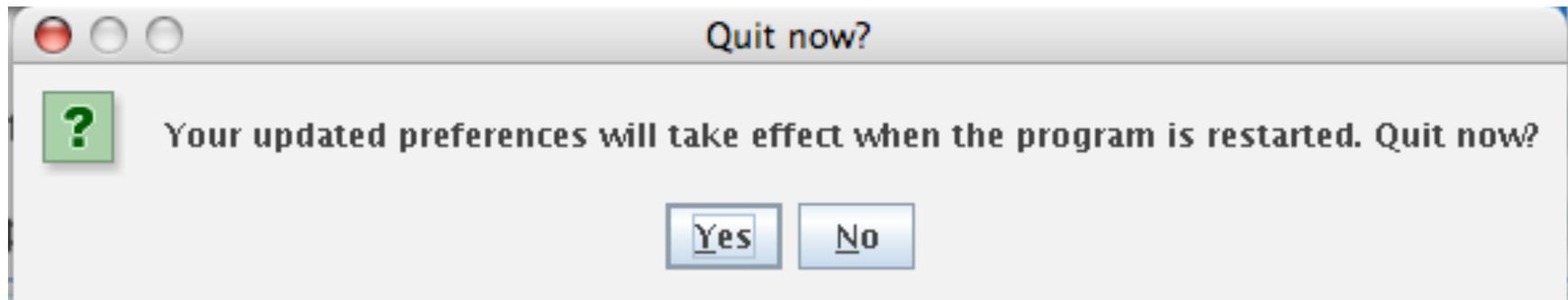


Click the “Save” button to write the connection configuration to disk.





Click the “Yes” button, to quit the program.
Restart the JMRI application.



Notes: Restart is required anytime preferences are changed for the preferences to take effect.

Preferences must be set for each JMRI application. They each have separate preferences files.



The program is set up according to the saved preferences.

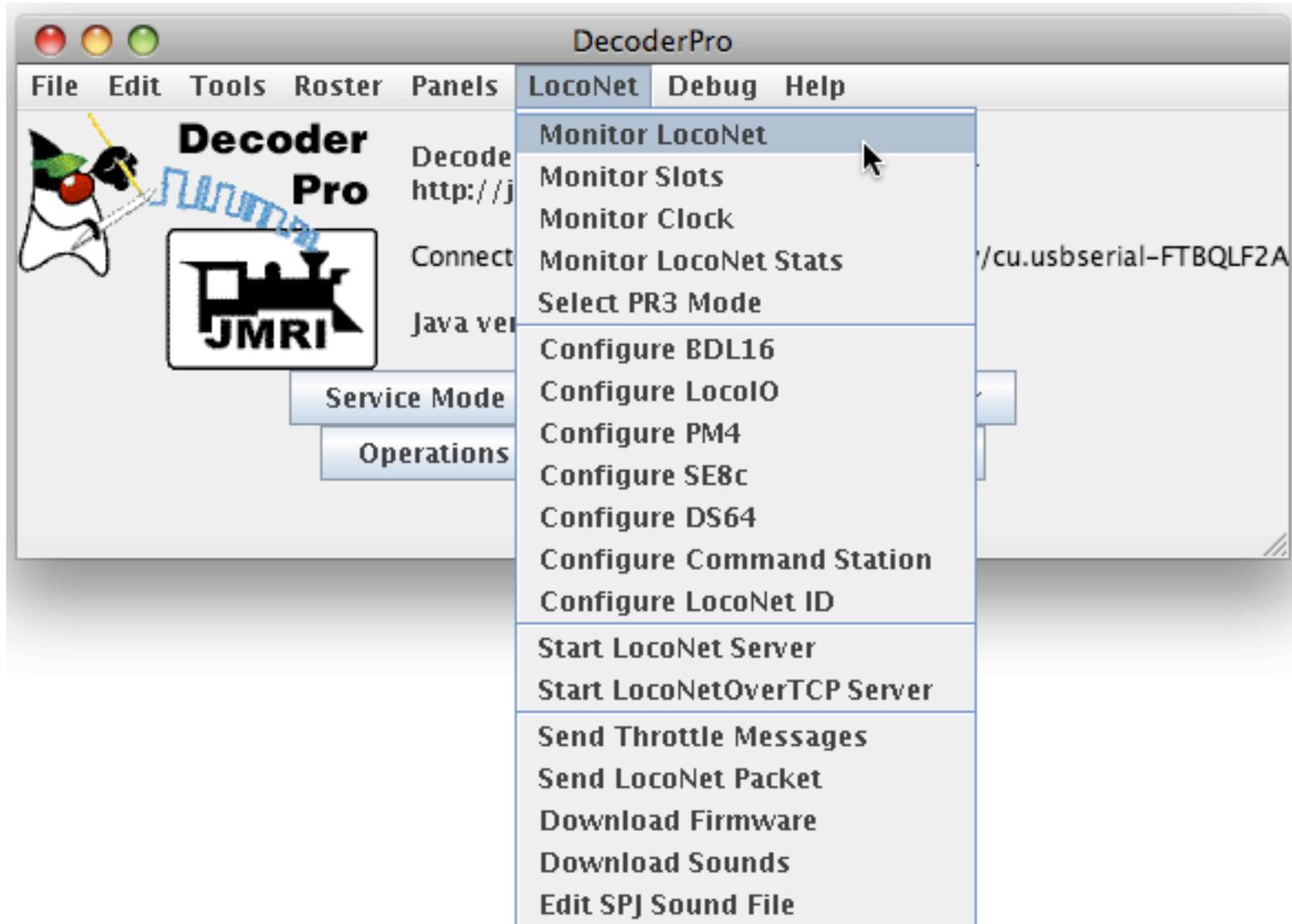


Note: Startup window contains program version and Java version, in addition to connection information.



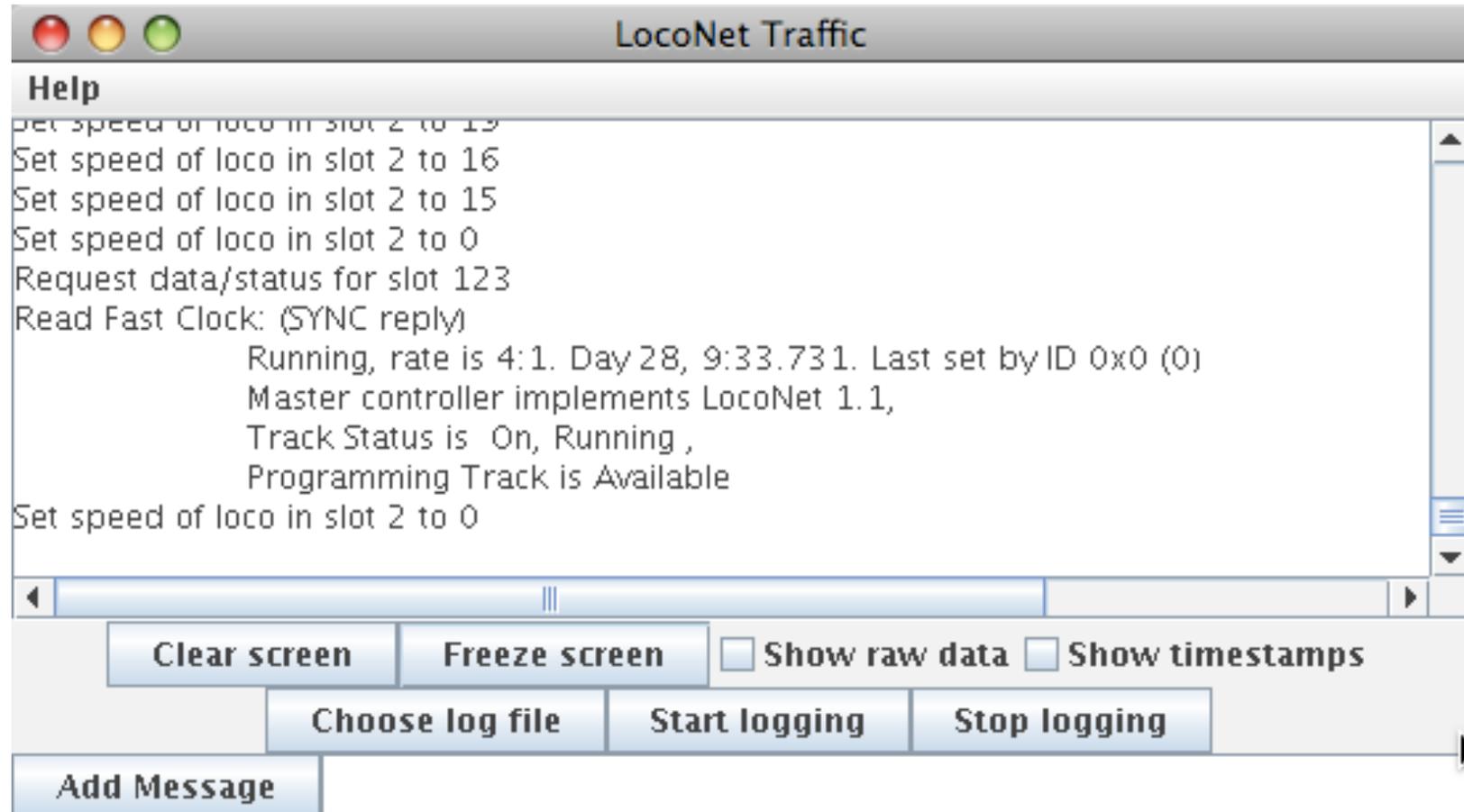
Connection Testing Example

Select “Monitor LocoNet” to snoop on traffic.



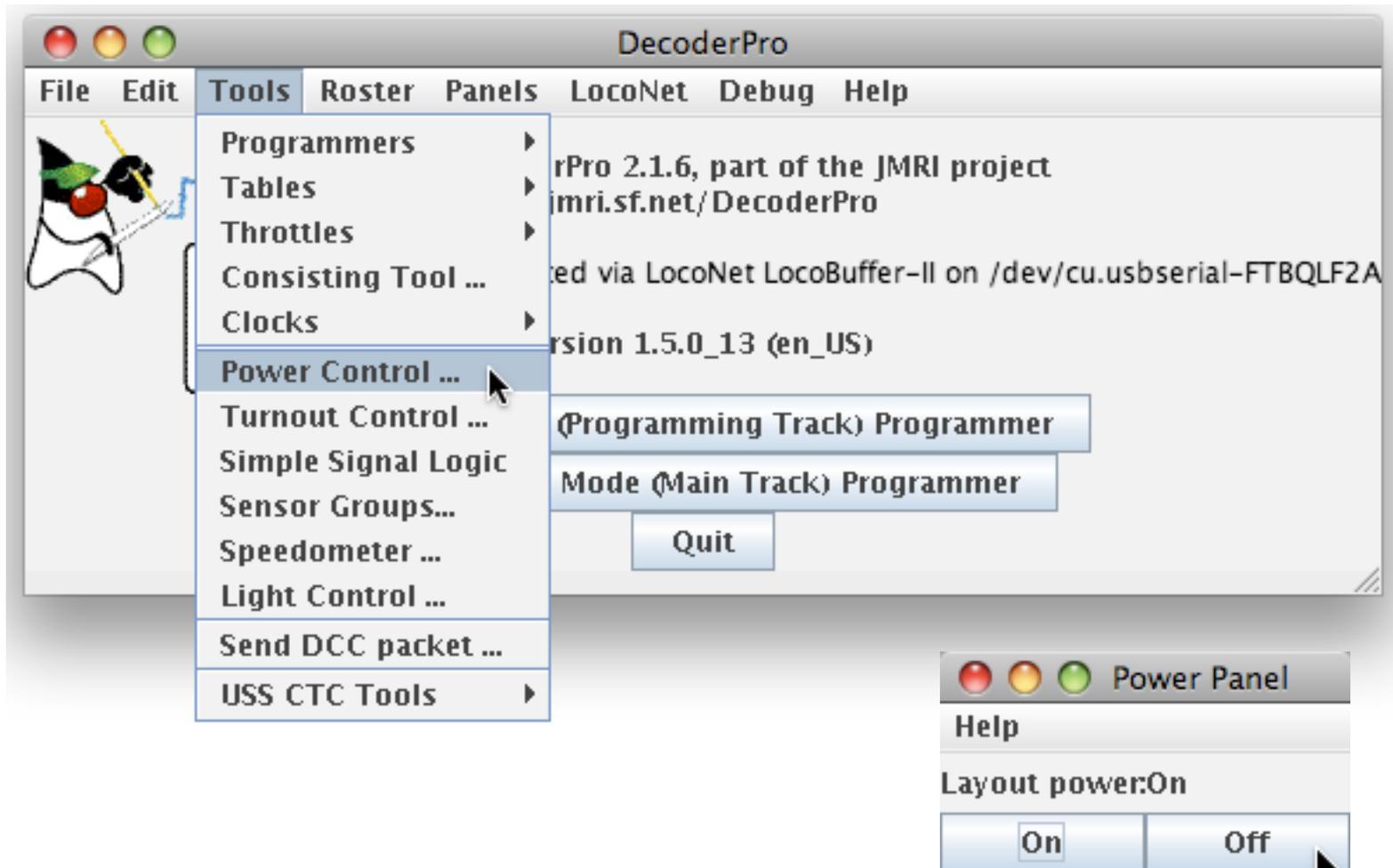


1. Plug in a throttle, and select a locomotive.
2. Look for traffic on the Loconet.



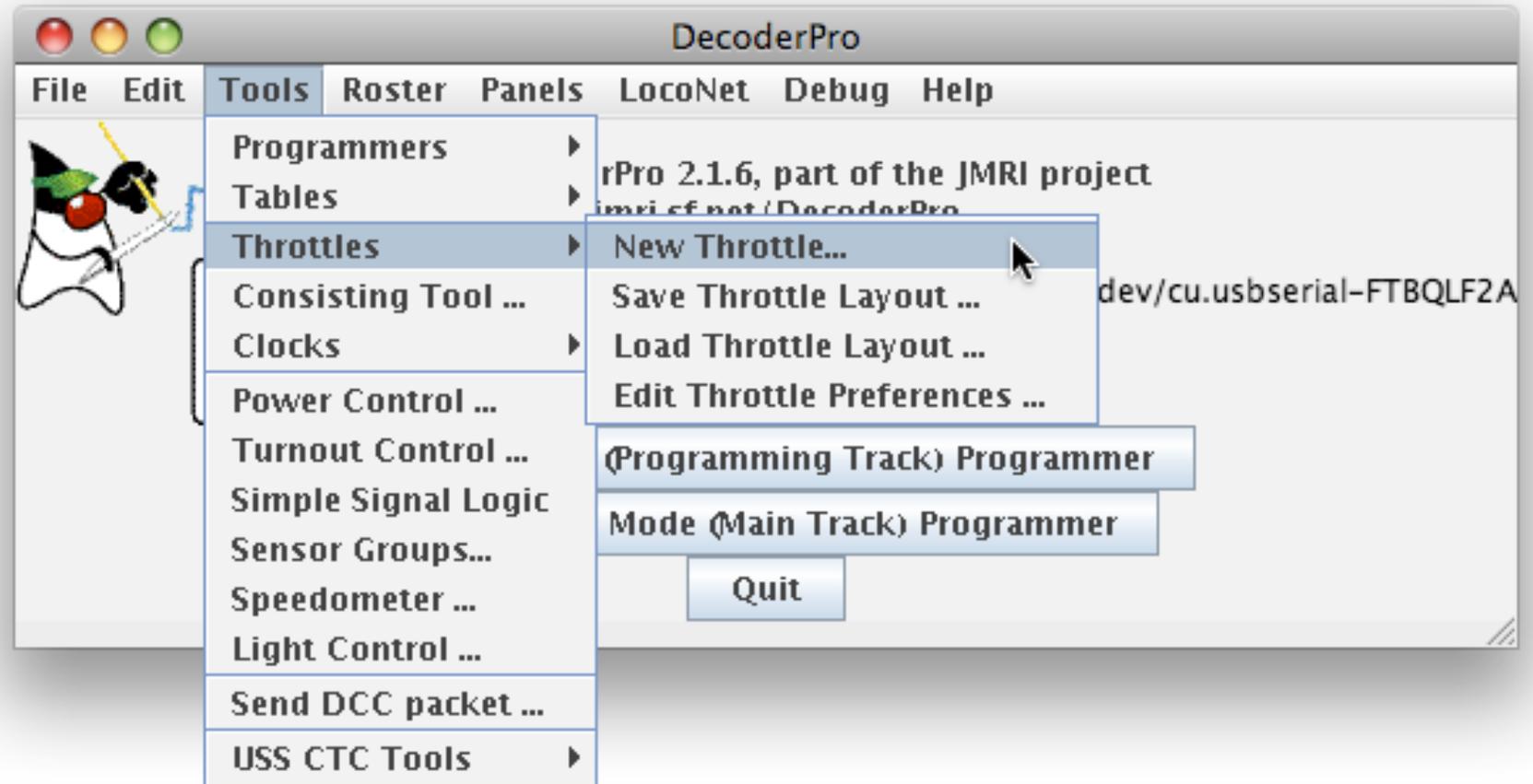


1. Select **Power Control...**
2. See if you can turn track power on/off.





If traffic looks reasonable and power goes on/off.
Open a JMRI throttle to test further.



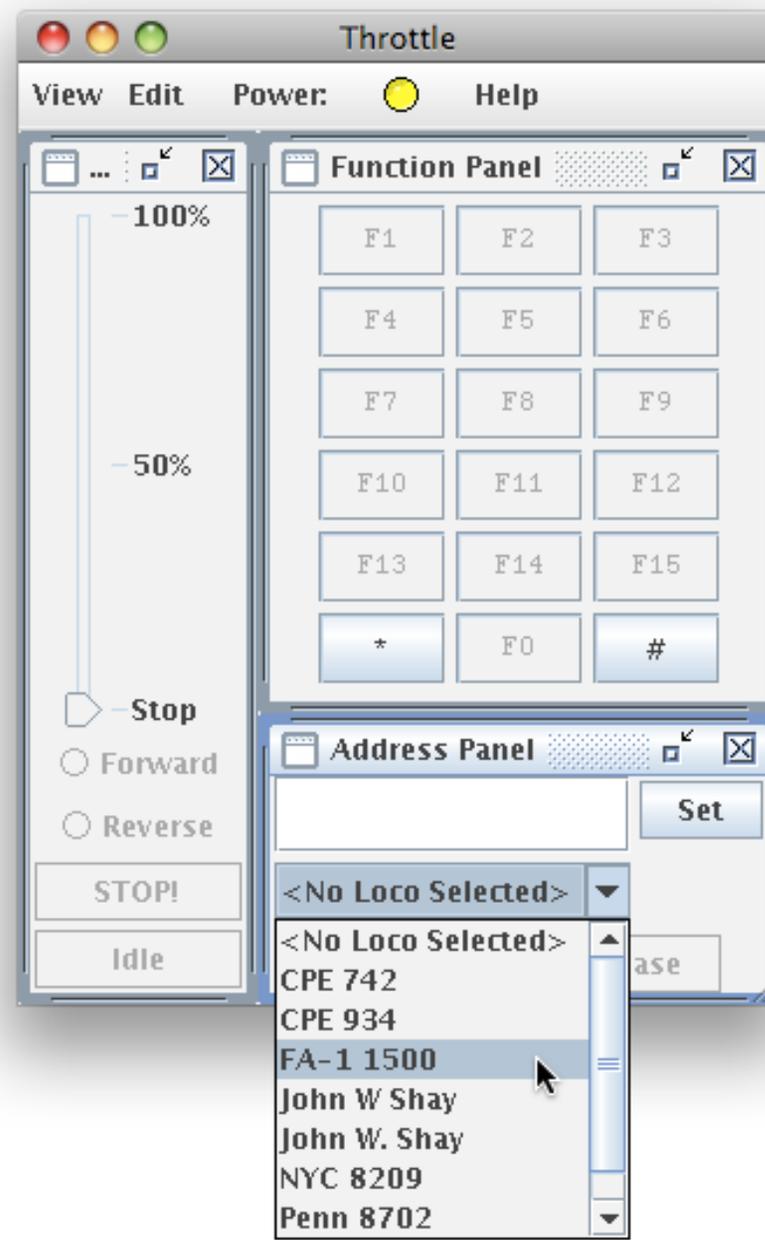


Select a train from
the Roster.

- or -

Type in a locomotive
address.

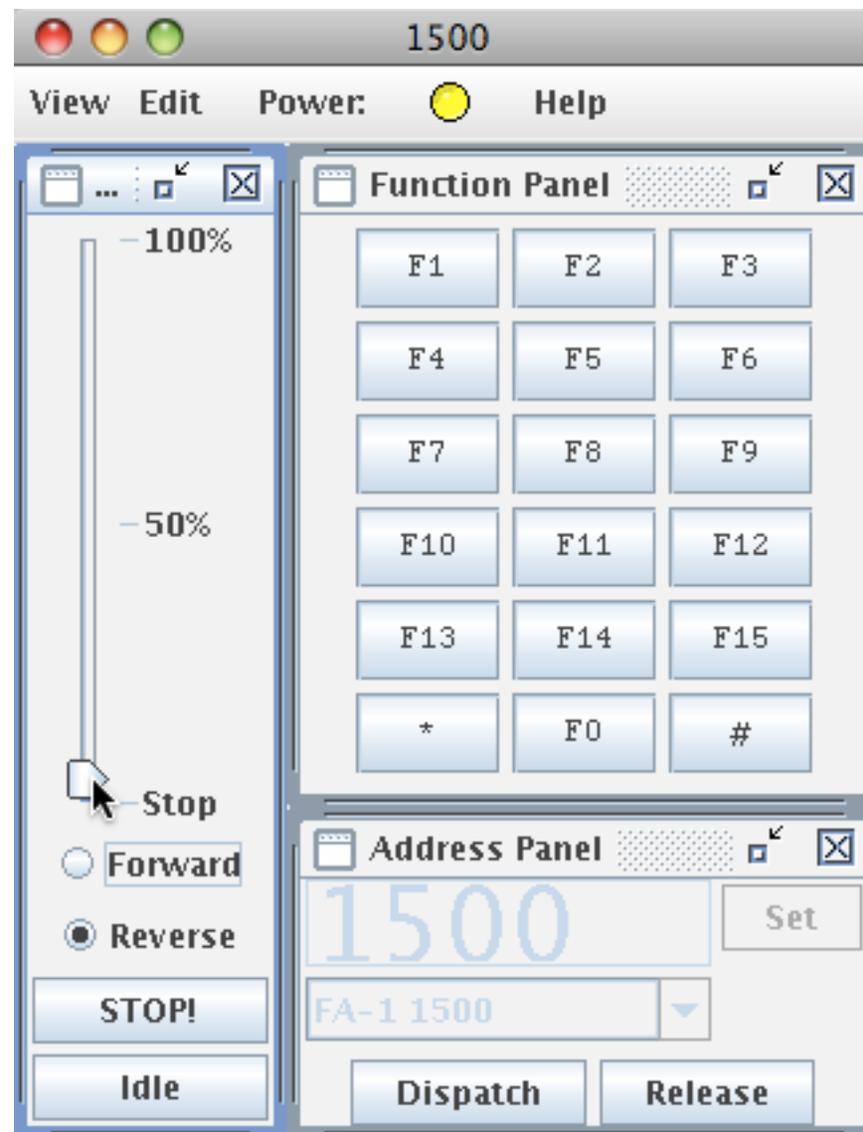
Click the “Set”
button.





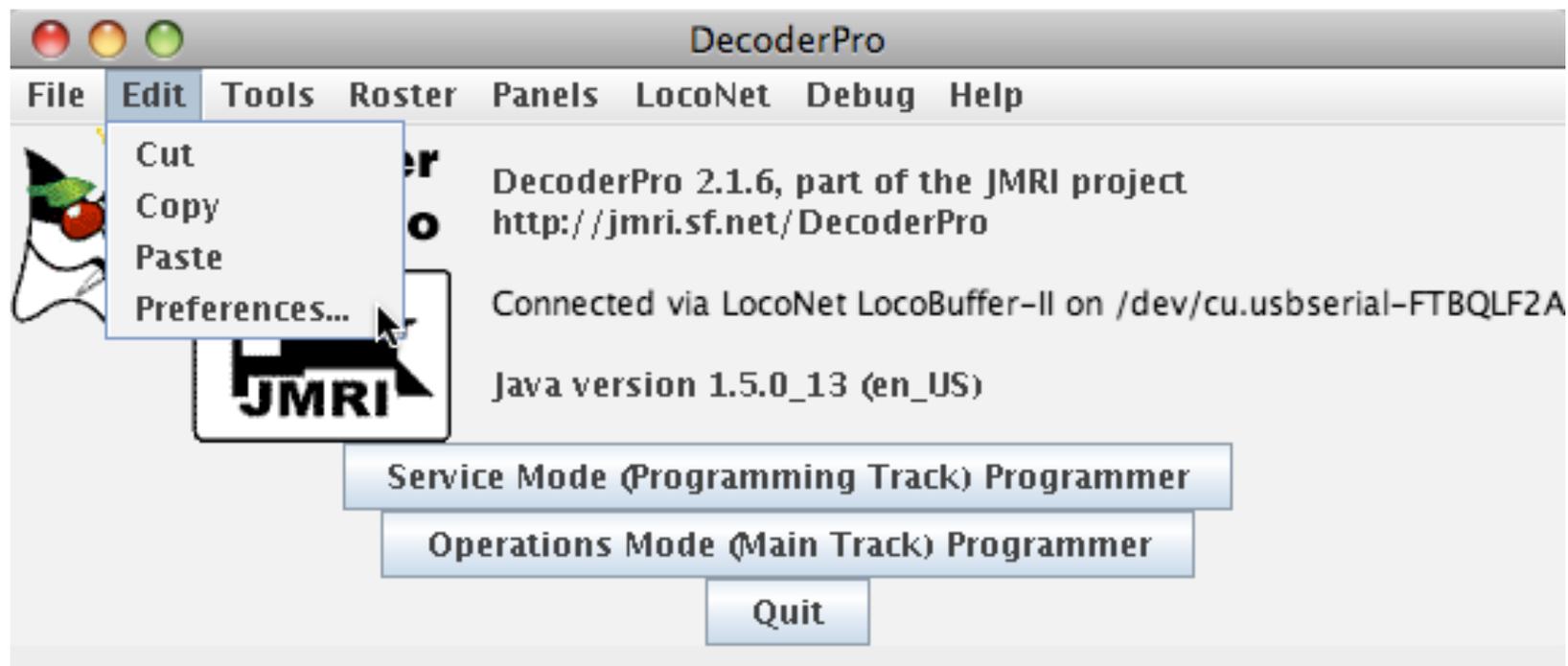
Run a train from
the computer.

If all works OK,
then JMRI is
successfully
communicating
with your
command station.





Configuration preferences may be accessed at any time via the Edit menu.





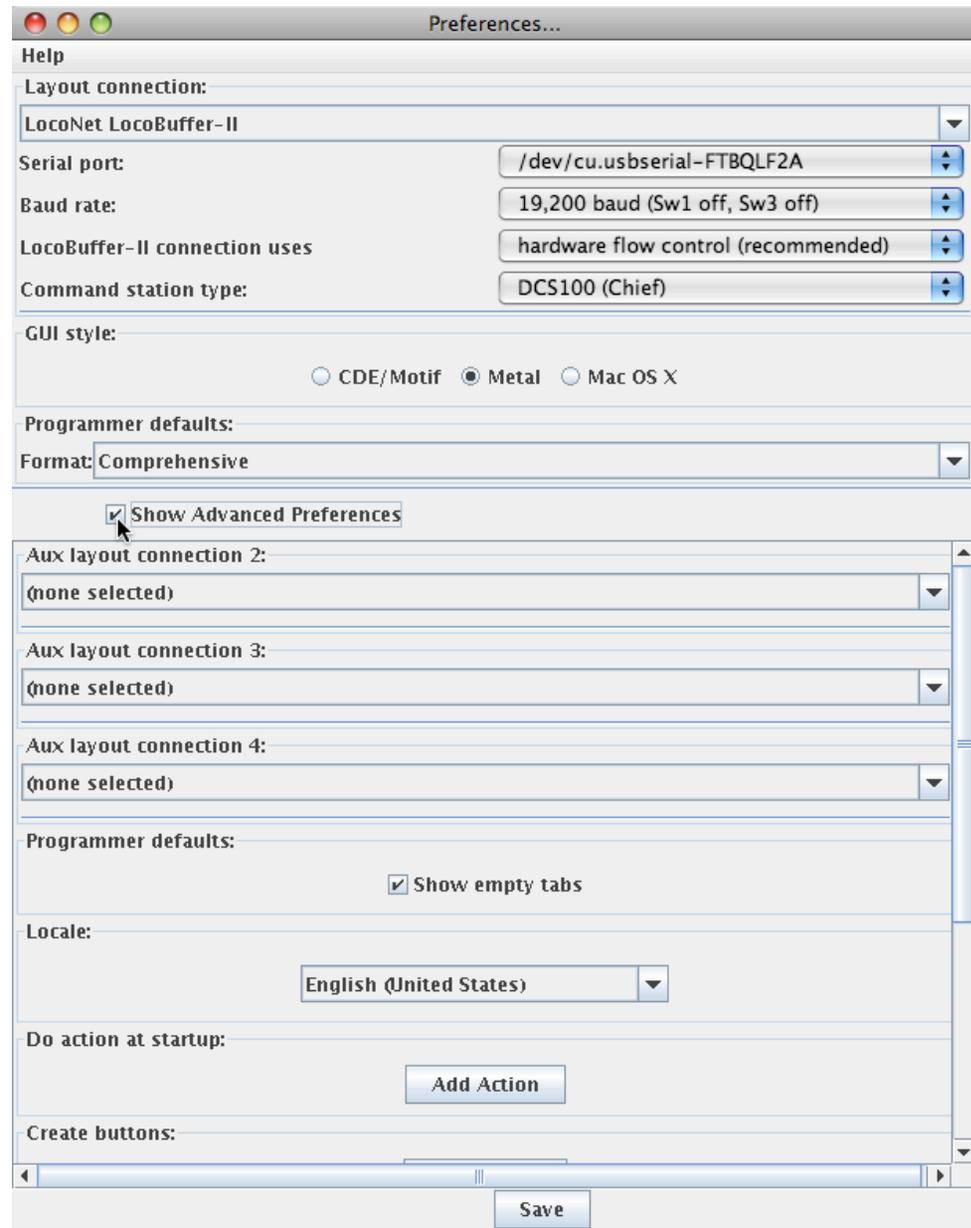
Advanced Preferences

Allow many useful options including:

A second layout connection.

Automatic loading files at startup.

Running scripts at startup.





How do I get help?

1st - Most JMRI windows have a Help menu.

Window Help ... Documentation related to that window

General Help ... Overall JMRI documentation

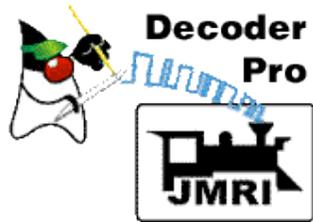
2nd - The JMRI web site - [http://
jmri.sourceforge.net/](http://jmri.sourceforge.net/)

Documentation and detailed instructions

3rd - JMRI Yahoo discussion group. **jmriusers**

Monitored by JMRI 'experts', eager to provide help.

Information on JMRI web site on how to sign up.



What is DecoderPro?

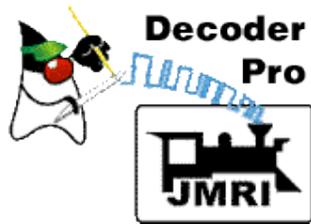
DecoderPro is a JMRI application.

DecoderPro is a better tool for programming DCC decoders.

DecoderPro simplifies the job of configuring complicated DCC decoders.

DecoderPro supports mobile decoders (decoders in locomotives).

DecoderPro supports some static decoders.



Basic Terminology

Decoder - small microcomputer based control unit

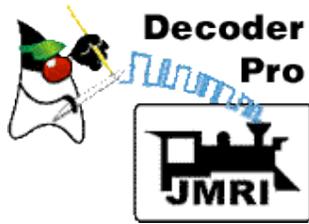
Mobile Decoder - Decoder in a locomotive,
“decodes” DCC commands to control locomotive.

CV (Control Variable) - 8-bit data byte in a
decoder that specifies user options.

Programming a Decoder - setting the values of the
CV's to user's options.

Decoders have many CV's. Many CV's follow
NMRA Standards, but some are vendor specific.

Each mobile decoder has an **Address** - a number that
allows the locomotive to be uniquely identified.



Setting up an Address

Decoder (locomotive) addresses can be 2 digits or 4 digits on modern decoders and DCC throttles.

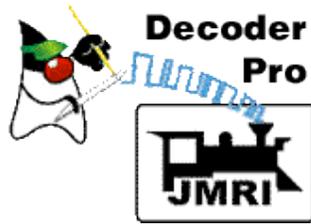
Usually set the address to the locomotive number.

Most decoders are set to address 03 on arrival.

A locomotive will respond to speed control and function commands that bear its address.

Setting the address is usually the first (and sometimes the only) programming needed.

It's easy to set up an address in DecoderPro.

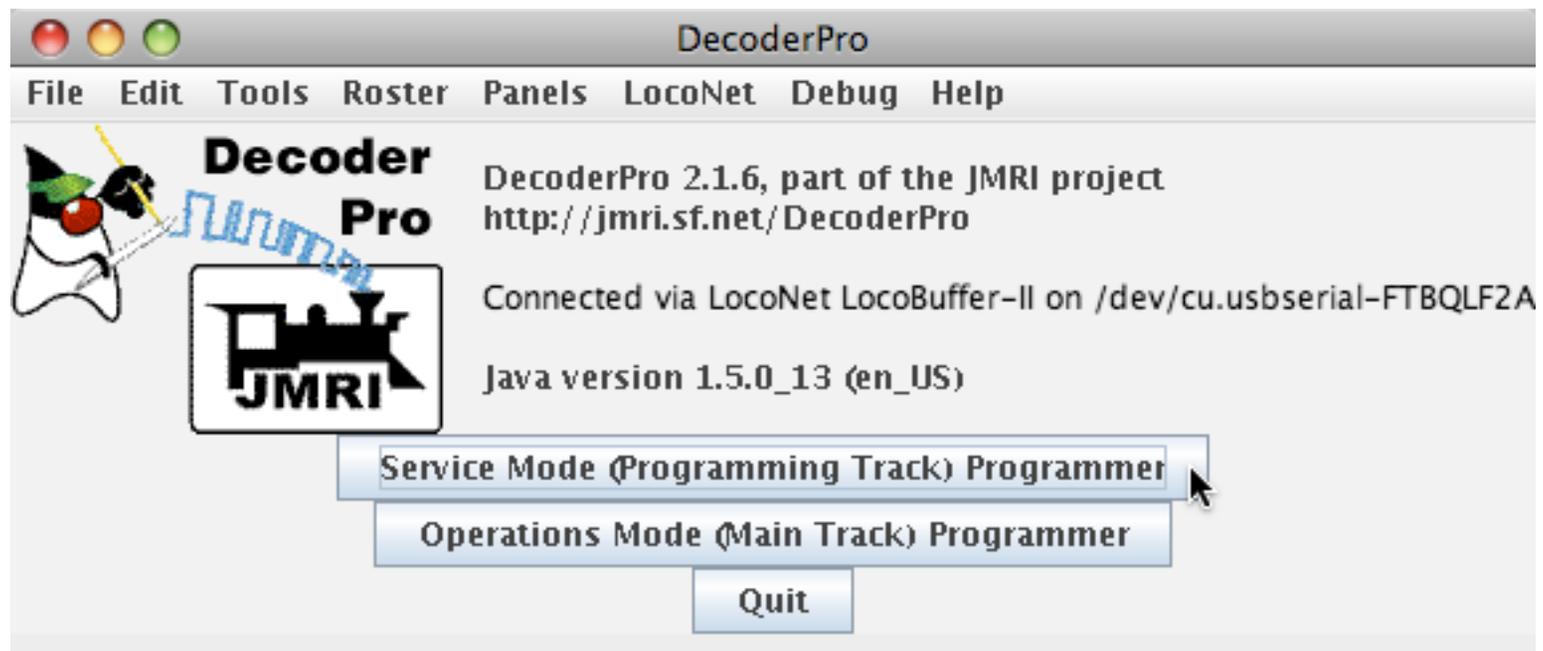


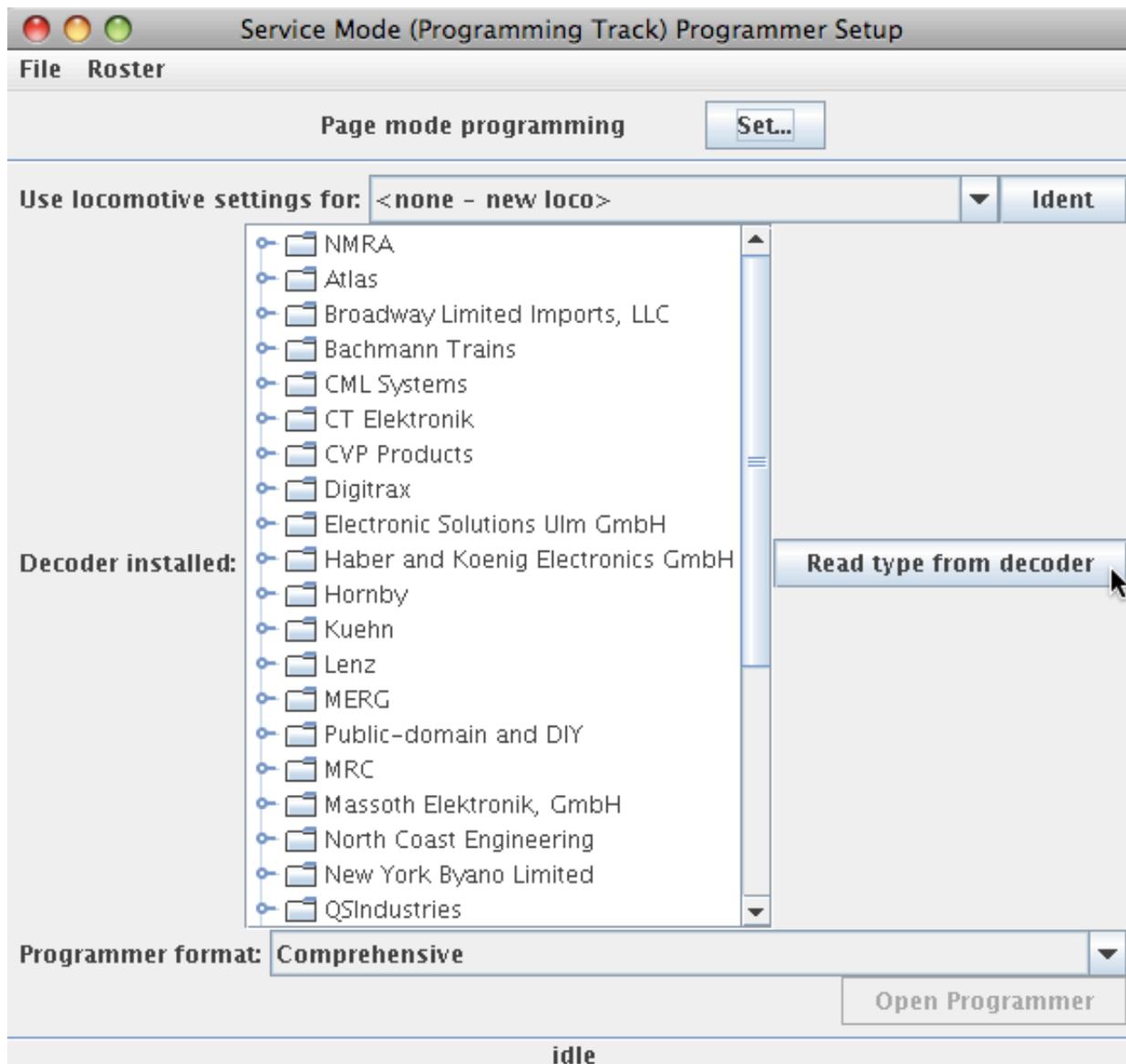
Example - Setting the address of a new decoder

Put the locomotive with the new decoder on the programming track.

Start Decoder Pro. When the window below comes up, click

“Service Mode (Programming Track) Programmer”





NMRA standards:

Two CV's identify
a decoder:

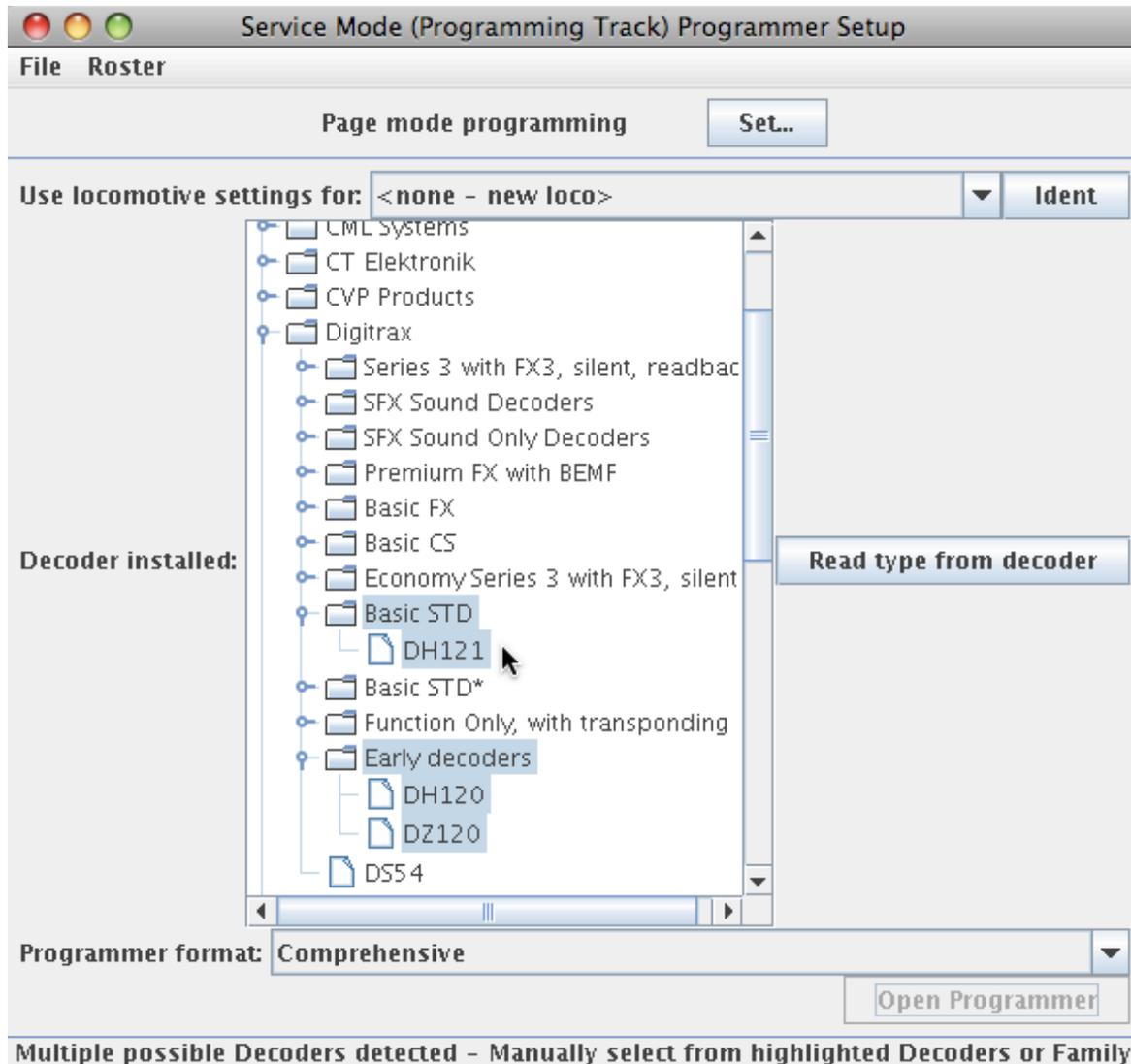
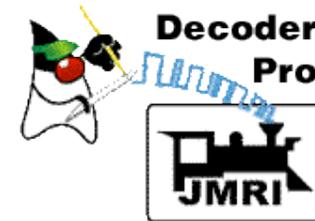
CV8 - Manufacturer ID

CV7 - Manufacturer
Version Number.

Both are **read only**.

<- Click here to have
DecoderPro attempt to
identify the decoder by
reading these CV's.

**Note: Some command
stations cannot read
CV's! For these,
select the decoder
in the list manually.**

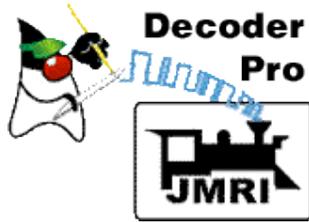


DecoderPro identified the decoder as a Digitrax DH121 with others possible.

(Sometimes the user has to choose among several possibilities.)

Click on DH121, to select it, and click “Open Programmer”.

<-



Fill in Roster information and click “Save”.

Program <new loco> in Service Mode (Programming Track)

File Reset

Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map Lights

ID: <new loco>

Road Name:

Road Number:

Manufacturer:

Owner:

Model:

DCC Address: --- ▾

Comment:

Decoder Family: Basic STD

Decoder Model: DH121

Decoder Comment:

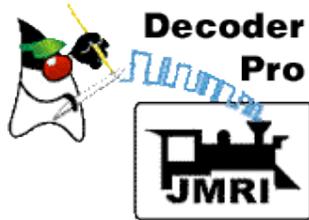
Filename:

Save Reset to defaults

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set..

idle



Click the Basic tab.

Program <new loco> in Service Mode (Programming Track)

File Reset

Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map Lights

ID: FA-1 1500

Road Name: UP 1500

Road Number: 1500

Manufacturer: Walther Trainline

Owner: Dave Duchamp

Model: ALCO FA-1

DCC Address: 3 Short

Comment:

Decoder Family: Basic STD

Decoder Model: DH121

Decoder Comment:

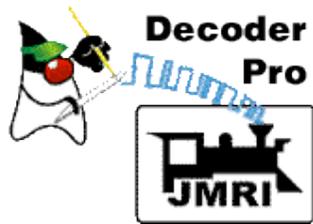
Filename:

Save Reset to defaults

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set..

Roster file FA_1_1500.xml saved OK



Click “Read full sheet”. Yellow items are replaced with factory default values.

Program <new loco> in Service Mode (Programming Track)

File Reset

Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map Lights

Active DCC Address: 3 One byte (short) address Two byte (extended) address

Primary Address 3 User Private ID #1 0

Long Address 0 User Private ID #2 0

Address Format One byte (short) address

Normal direction of motion forward

Speed steps 28 speed step format

Analog (DC) Operation DC conversion enabled

Manufacturer ID 0

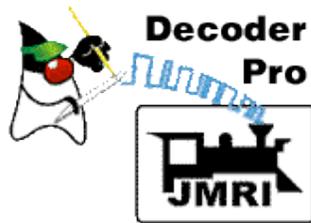
Manufacturer Version No 0

Read changes on sheet Write changes on sheet Read full sheet Write full sheet

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set..

Roster file FA_1_1500.xml saved OK



Switch off analog, and set new two-byte address.
Click “Write changes on sheet” to send to loco.

Program <new loco> in Service Mode (Programming Track)

File Reset

Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map Lights

Active DCC Address: 1500 One byte (short) address
 Two byte (extended) address

Primary Address 3 User Private ID #1 0
Long Address 1500 User Private ID #2 0
Address Format Two byte (extended) address ▼

Normal direction of motion forward ▼
Speed steps 28 speed step format ▼
Analog (DC) Operation NMRA Digital only ▼

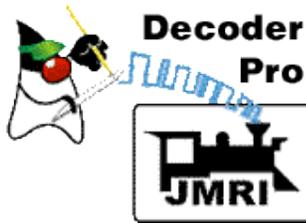
Manufacturer ID 129
Manufacturer Version No 34

Read changes on sheet Write changes on sheet Read full sheet Write full sheet

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set...

OK



Return to Roster Entry and “Save”
the Roster file to disk.

All done!

Program <new loco> in Service Mode (Programming Track)

File Reset

Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map Lights

ID: FA-1 1500

Road Name: UP 1500

Road Number: 1500

Manufacturer: Walthers Trainline

Owner: Dave Duchamp

Model: ALCO FA-1

DCC Address: 3 Short

Comment:

Decoder Family: Basic STD

Decoder Model: DH121

Decoder Comment:

Filename:

Save Reset to defaults

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set..

Roster file FA_1_1500.xml saved OK



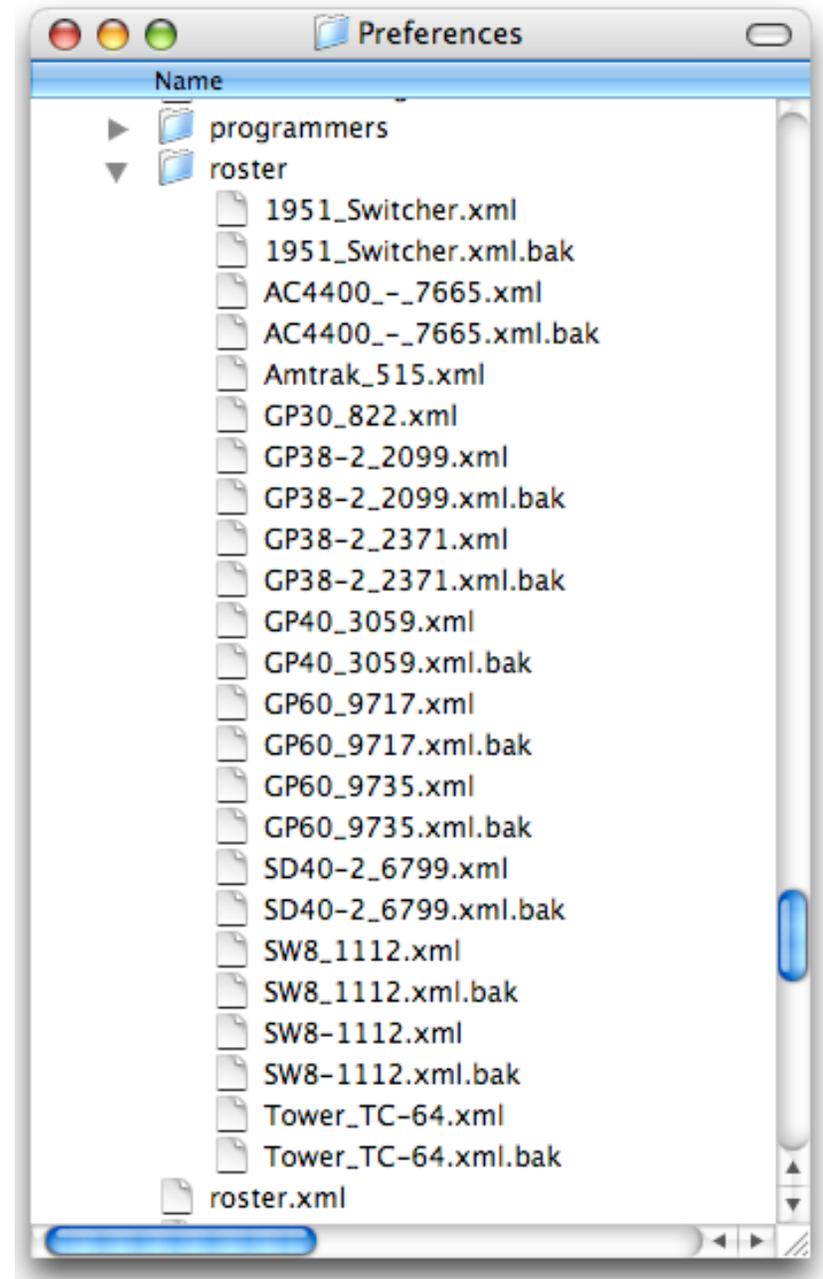
What are Roster Files?

DecoderPro stores the final information for each decoder in a **Roster File**.

These Roster Files are used to construct a Roster menu for JMRI applications.

A Roster file allows easy reprogramming if decoder needs to be reset.

The Roster menu allows easy selection of a loco in JMRI tools--decoder programmer, throttle, consist, etc.

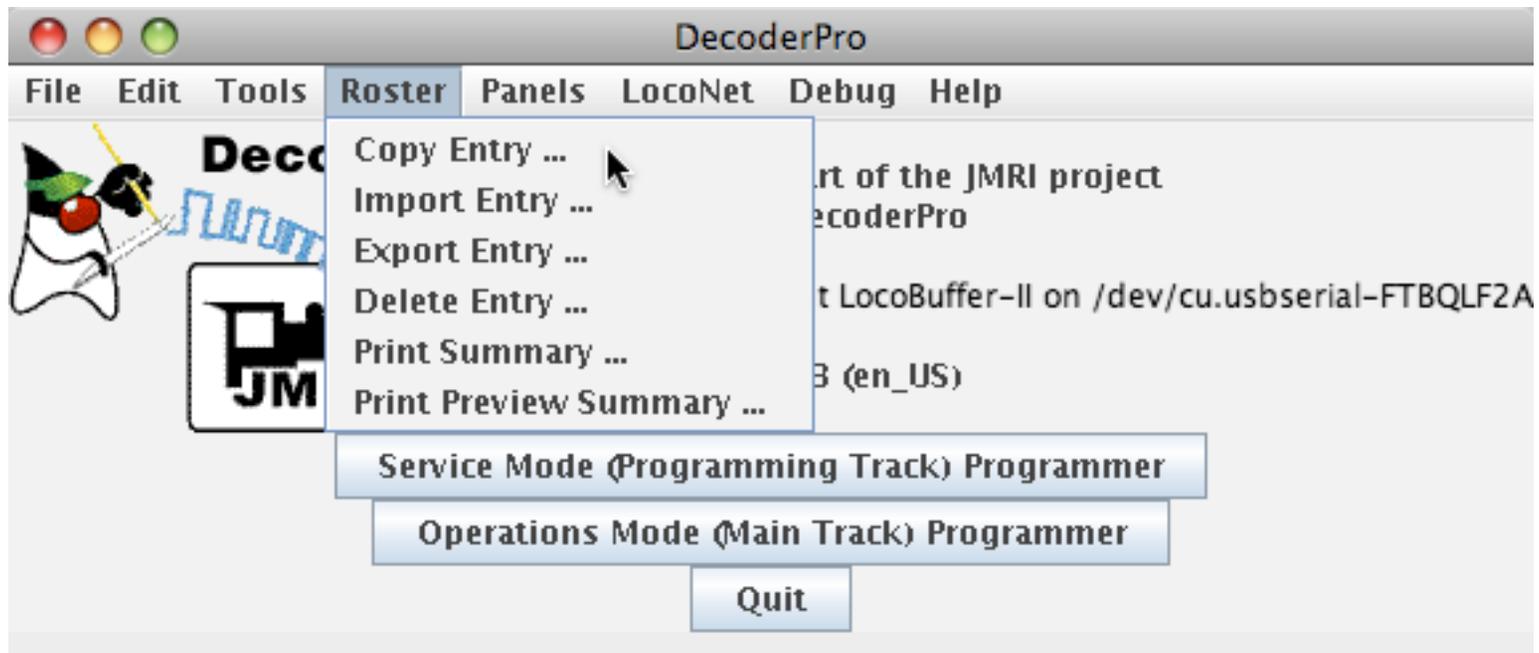


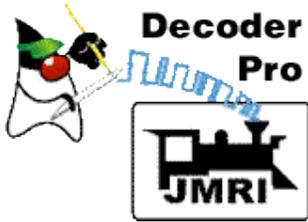


The Roster Menu

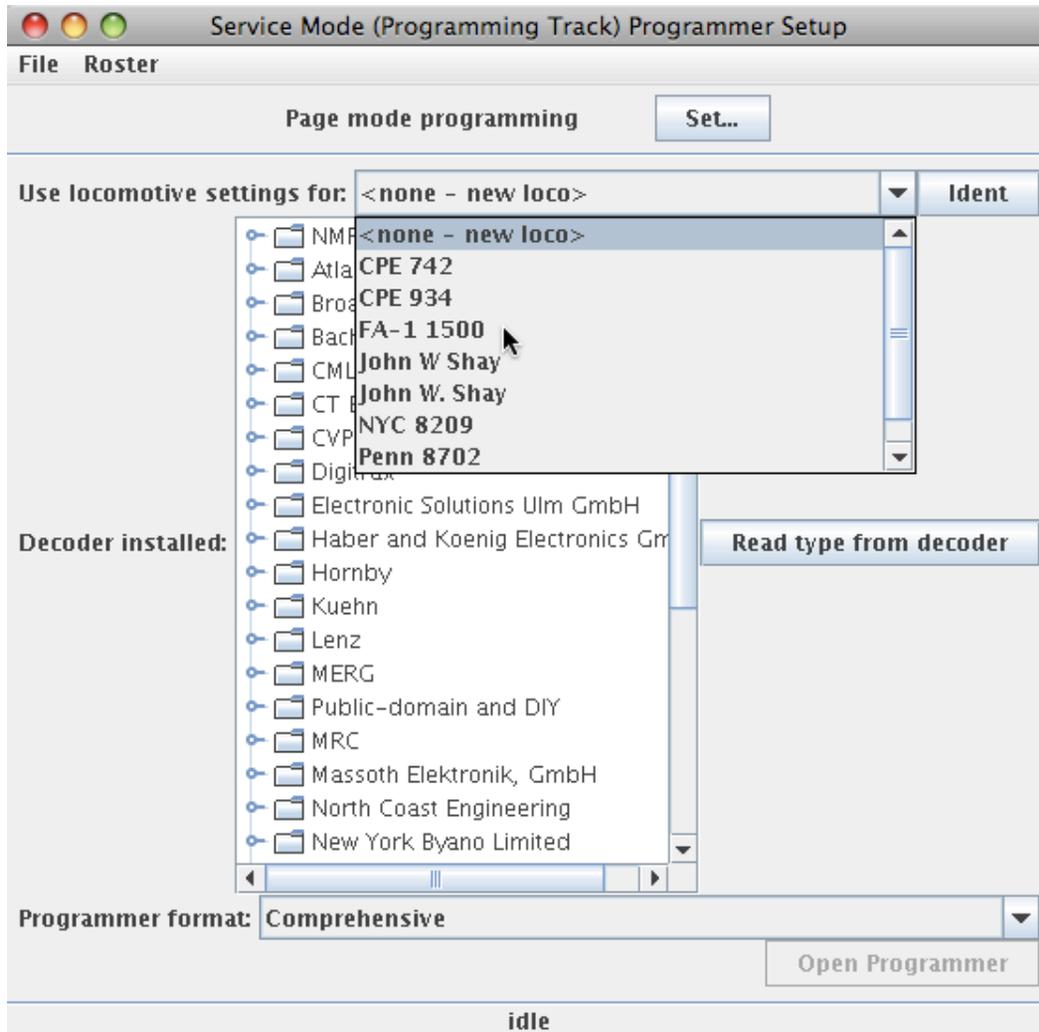
Provides useful functions when working with your Rosters.

Accessed in Programmer Setup window and in main DecoderPro window.





Changing a decoder's programming



Select loco from Roster

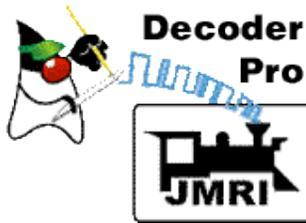
- or -

<- Click “Ident” to have DecoderPro read the loco address and find it in the Roster.

After loco is identified, click “Open Programmer”

Note: “Open Programmer” is not active until a decoder is identified.

<-



The saved information is back!
Click “Speed Control”.

Program FA-1 1500 in Service Mode (Programming Track)

File Reset

Lights Analog Controls Consist Advanced Sound Sound Levels CVs

Roster Entry Basic Motor Speed Control Function Map

ID: FA-1 1500

Road Name: UP 1500

Road Number: 1500

Manufacturer: Walthers Trainline

Owner: Dave Duchamp

Model: ALCO FA-1

DCC Address: 1500 Long

Comment:

Decoder Family: Basic STD

Decoder Model: DH121

Decoder Comment:

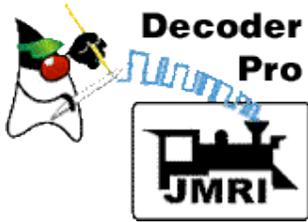
Filename: FA_1_1500.xml

Save Reset to defaults

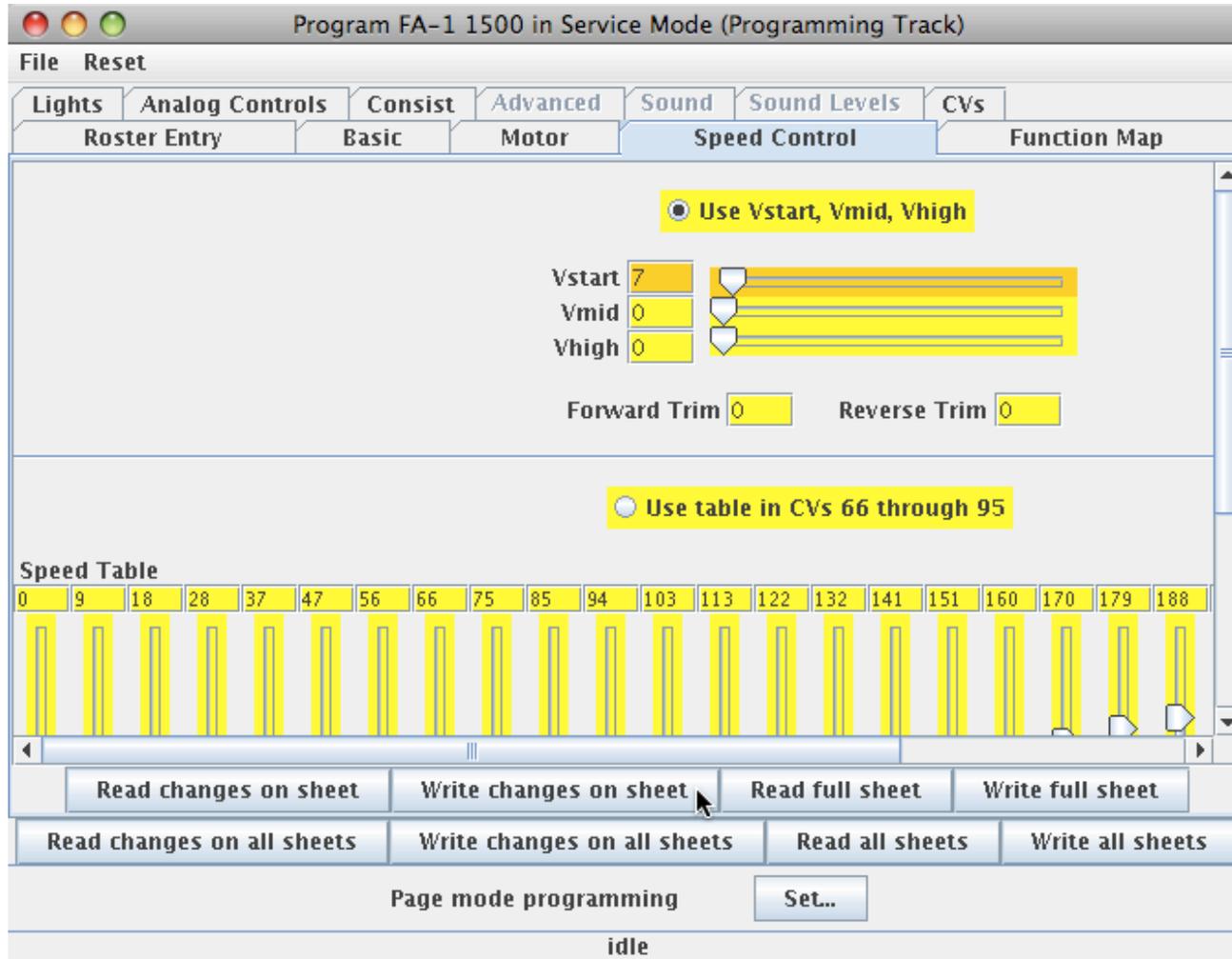
Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

Page mode programming Set..

idle



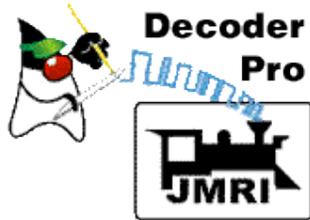
Yellow color indicates the values are from the Roster file. Orange - changed, but not written to decoder.



Enter a value in “Vstart”.

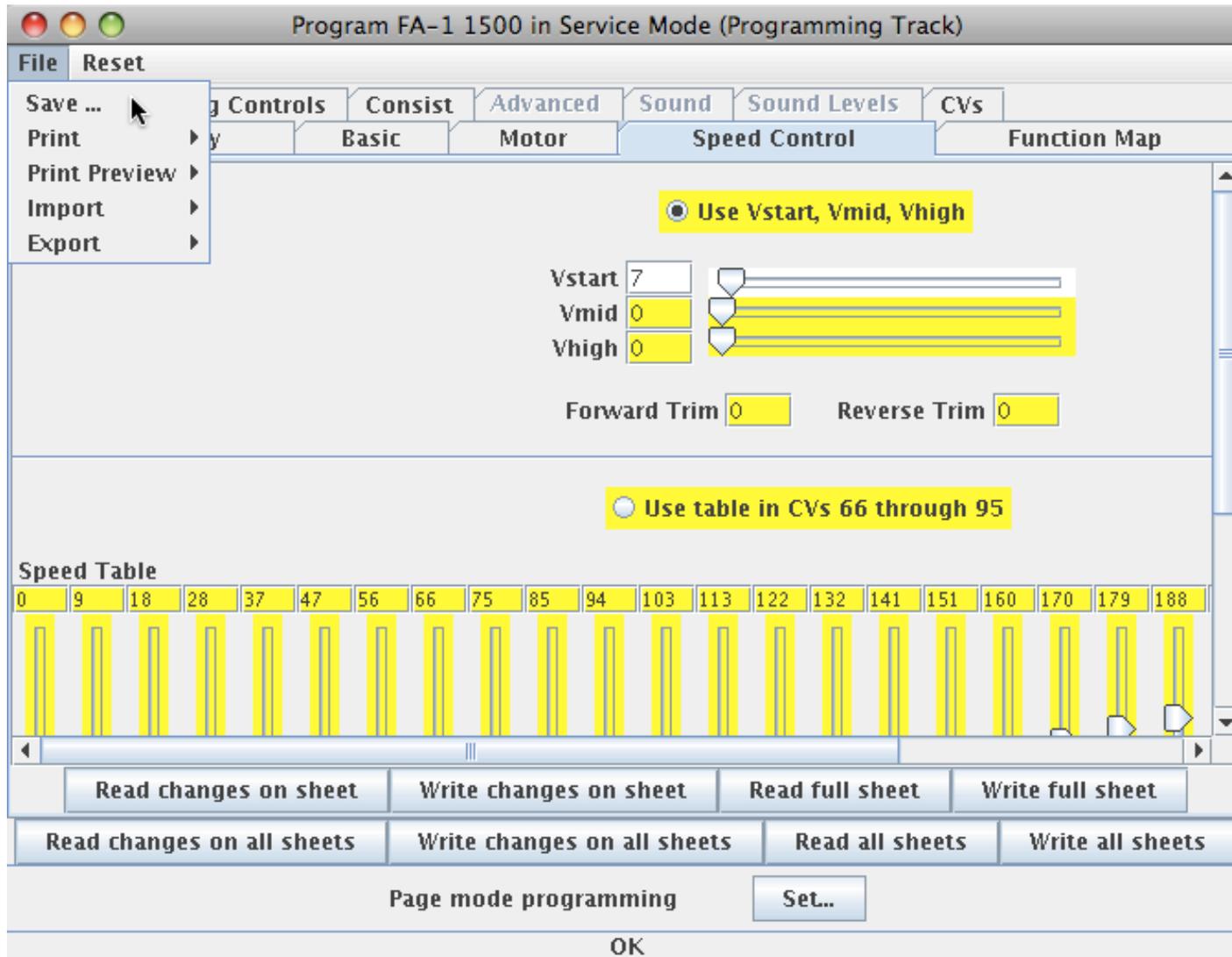
Click “Write changes on sheet”





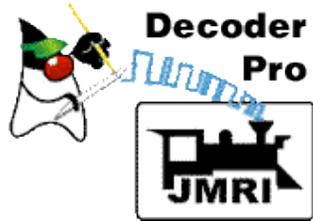
Changed Vstart was written to the decoder.

Select File>Save... to save the change to disk.



It's done.

Go run the loco!



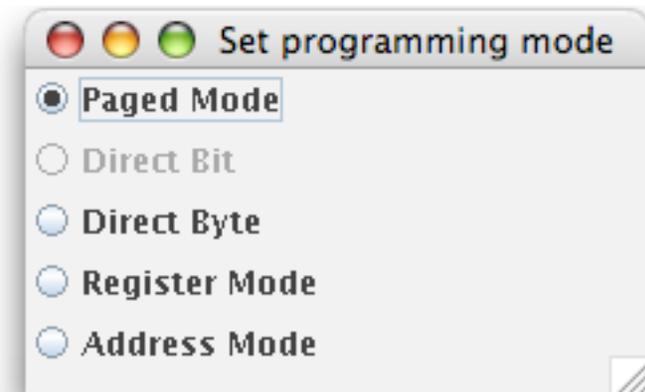
Miscellaneous Info and Tips

Support for new decoders is continuously added to DecoderPro.

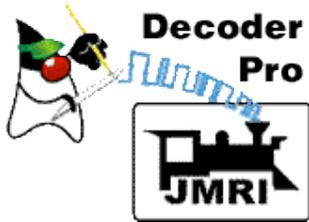
DecoderPro works through the command station, so it's usually limited to what you can do with your throttle. Think of DecoderPro as a smart throttle

DecoderPro supports other modes of programming. Access these other modes using the “Set...” button to get the dialog shown at the right.

Some decoders need a different mode for programming.



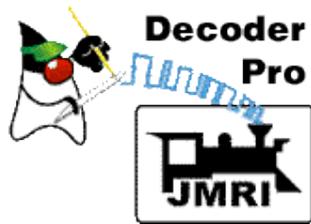
Some new sound decoders need a programming track booster to communicate with some command stations.



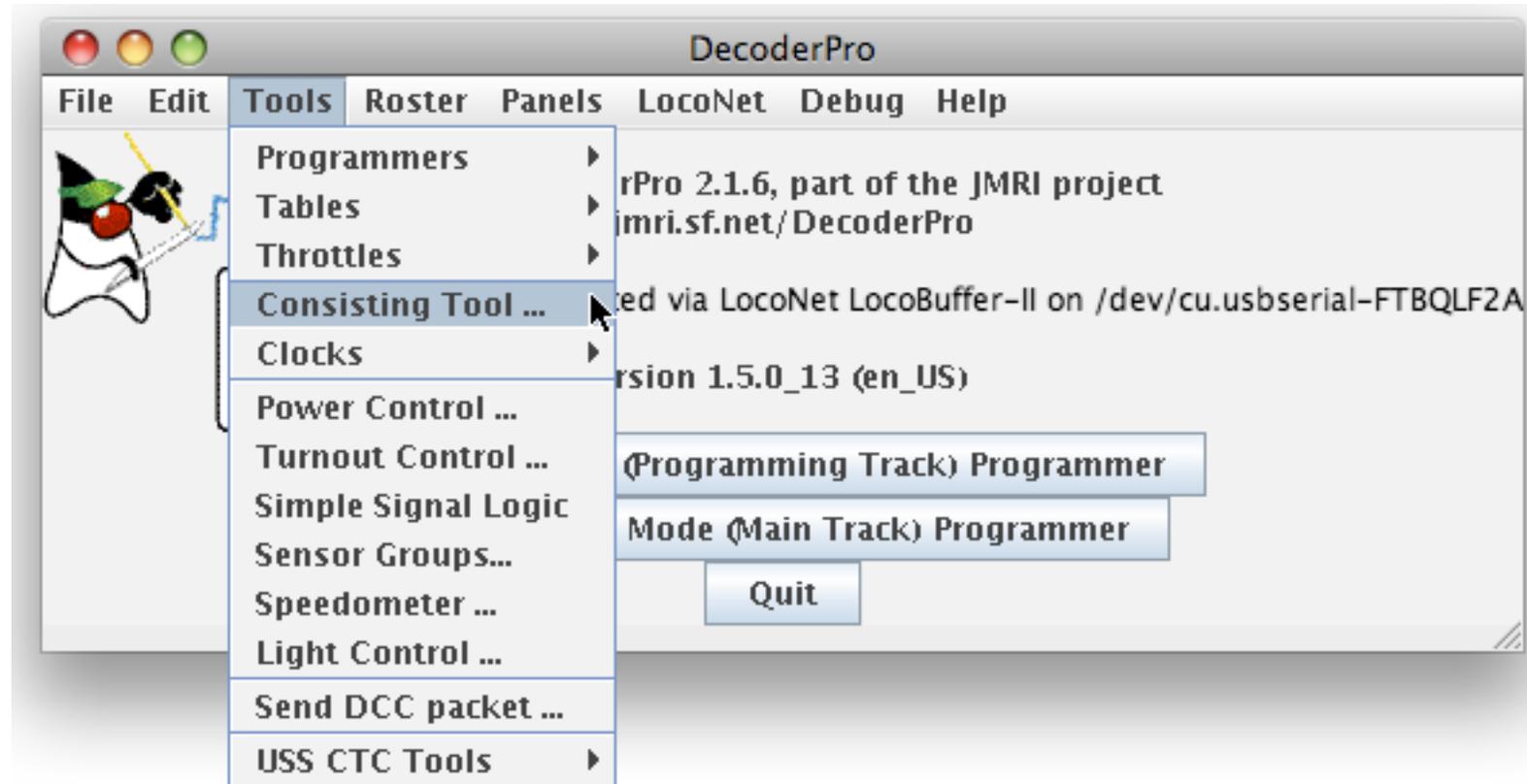
Example: Procedure for Speed Matching Engines for Consists

Object: To match the speed of two or more engines.

1. Preliminary: Determine which engine runs slowest. Warm up engines (3-4 minutes). Make sure wheels and track are clean!
2. Make sure all engines have DecoderPro roster files, and start speeds are set.
3. Make a consist with your slowest engine as the lead engine. **Do not couple the engines.**

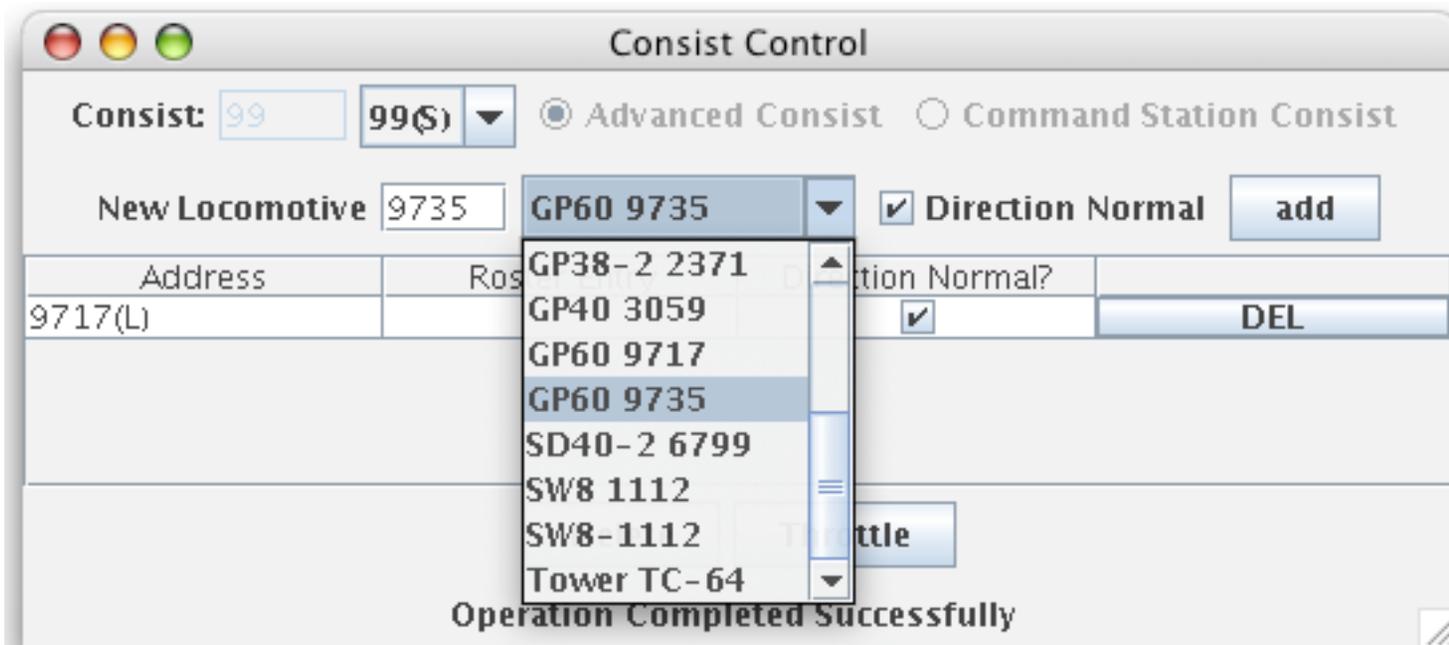


Select the JMRI Consisting Tool.



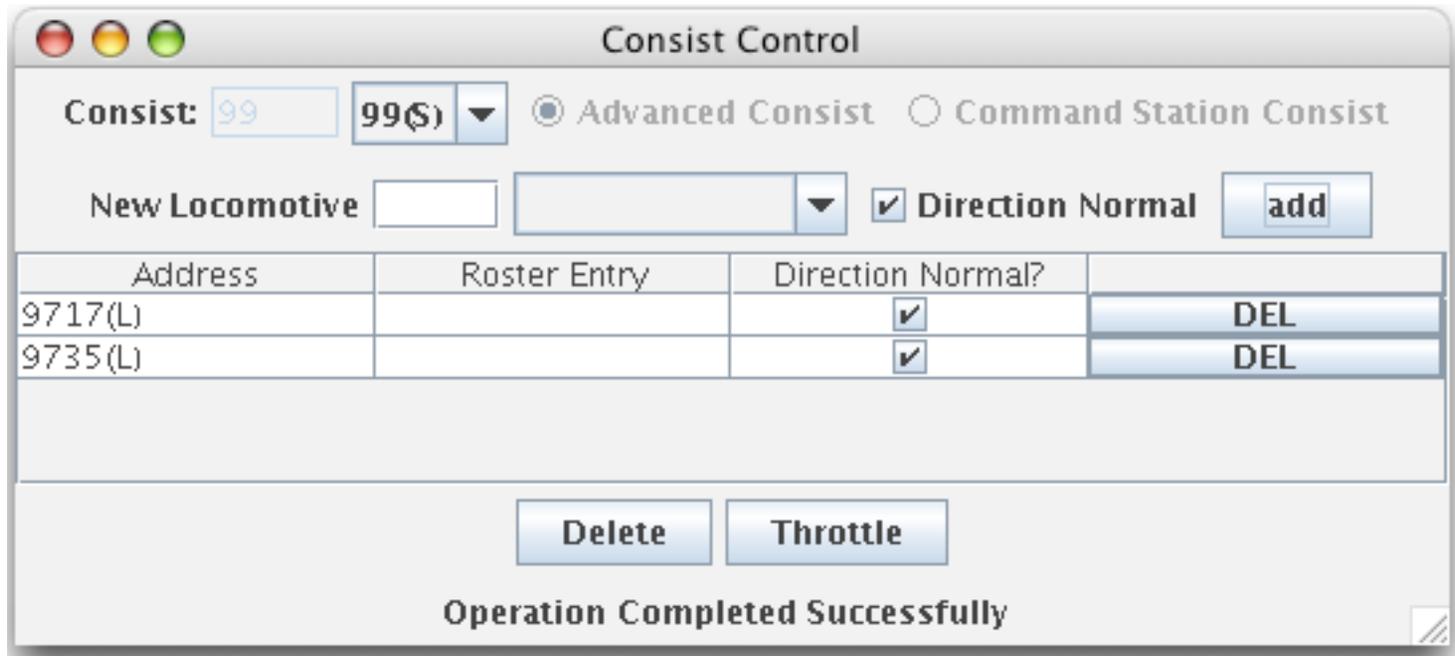


Give the new consist a two-digit address, and add the two engines to the table for the new consist, the slowest first.





After locomotives are in the table, hit the “Throttle” button to make the consist and open a new JMRI Throttle to control it.

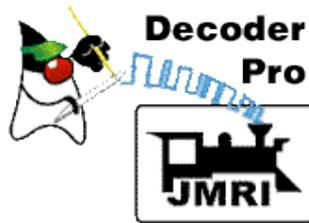


Address	Roster Entry	Direction Normal?	
9717(L)		<input checked="" type="checkbox"/>	DEL
9735(L)		<input checked="" type="checkbox"/>	DEL

Delete

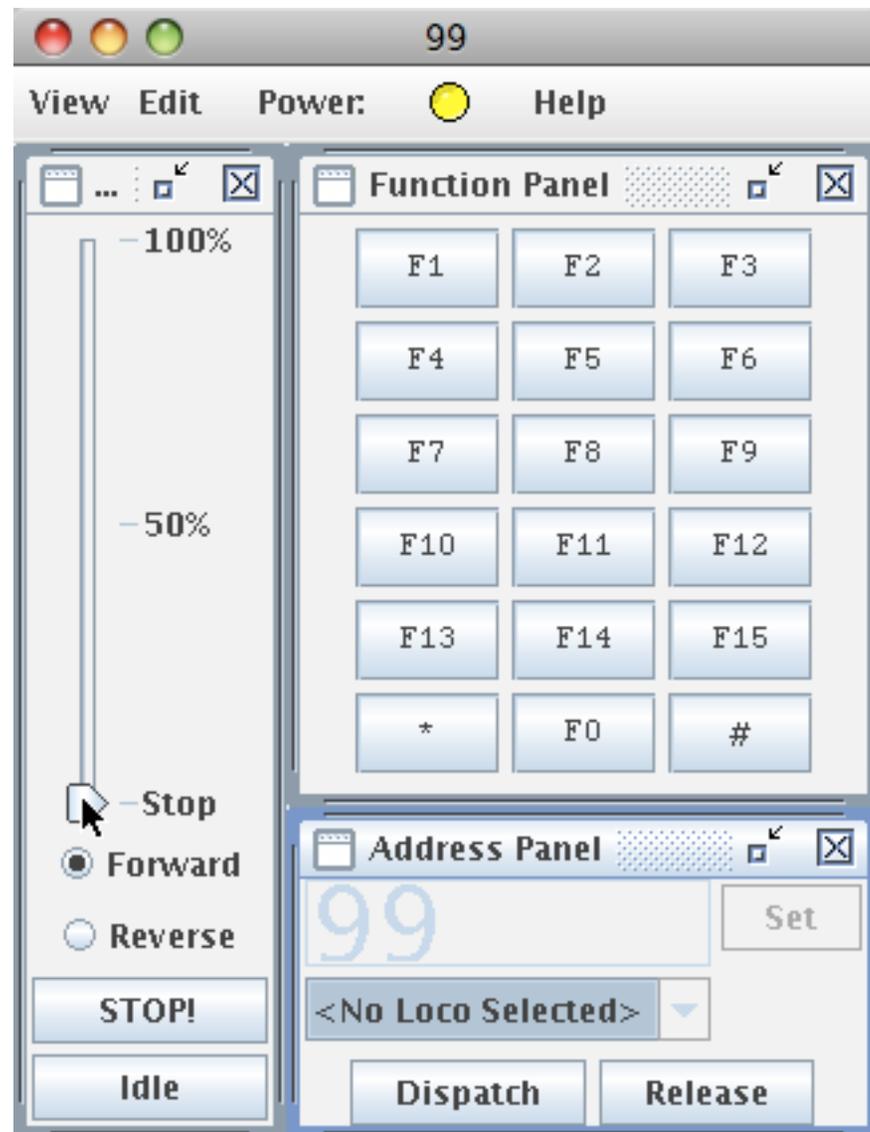
Throttle

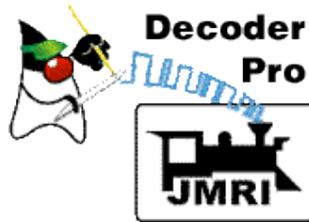
Operation Completed Successfully



You can now run
the consist using this
throttle.

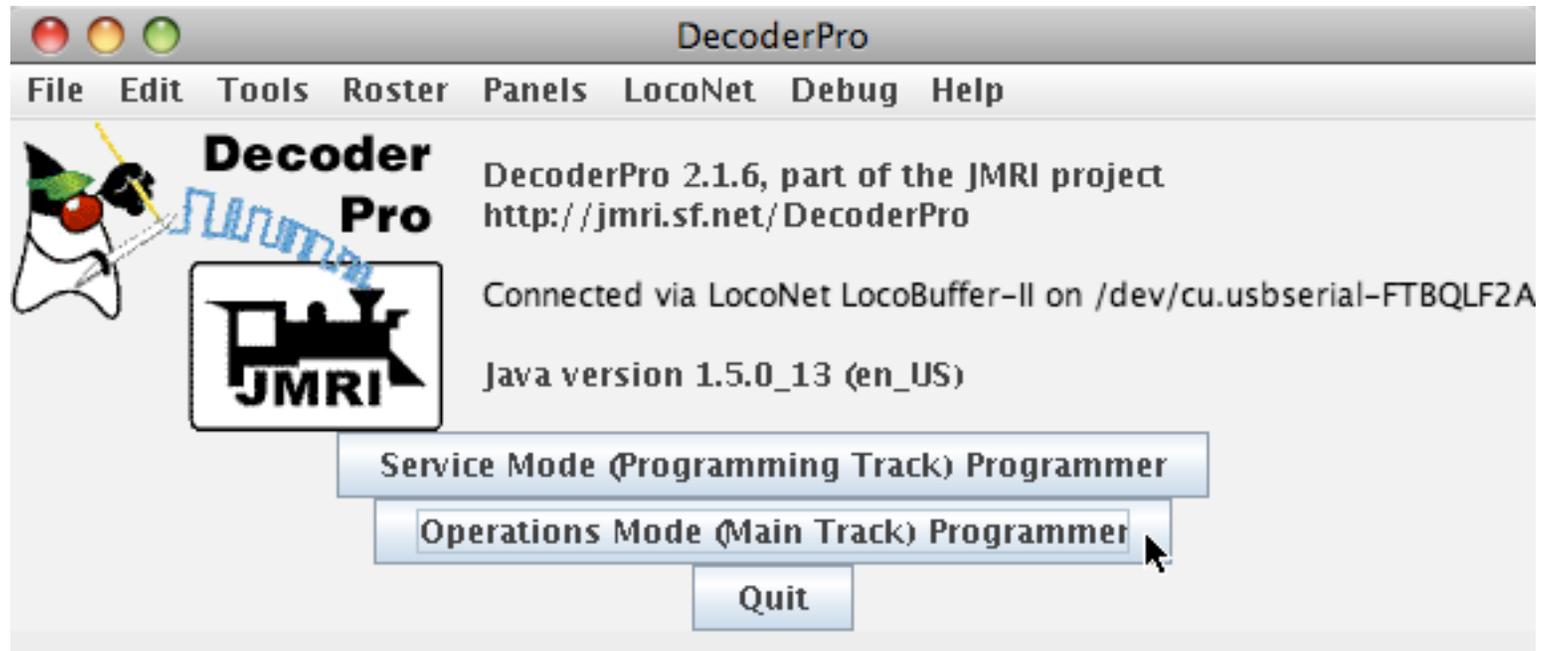
**Remember: Do not
couple the locos.
Space them about
12 inches apart.**

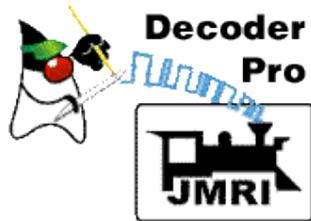




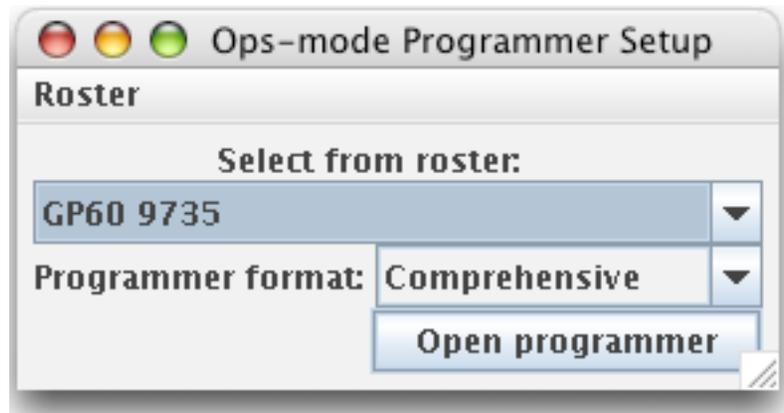
4. Open a programmer for the faster loco in DecoderPro using “Ops mode”.

Select “Operations Mode (Main Track) ...”





Select the roster entry for the faster loco, and click “Open Programmer”.





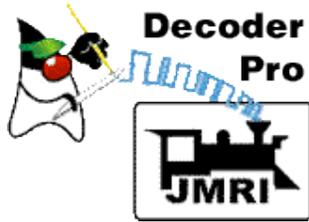
5. Slow the faster loco using the decoder speed table.

Select the “Speed Control” pane.

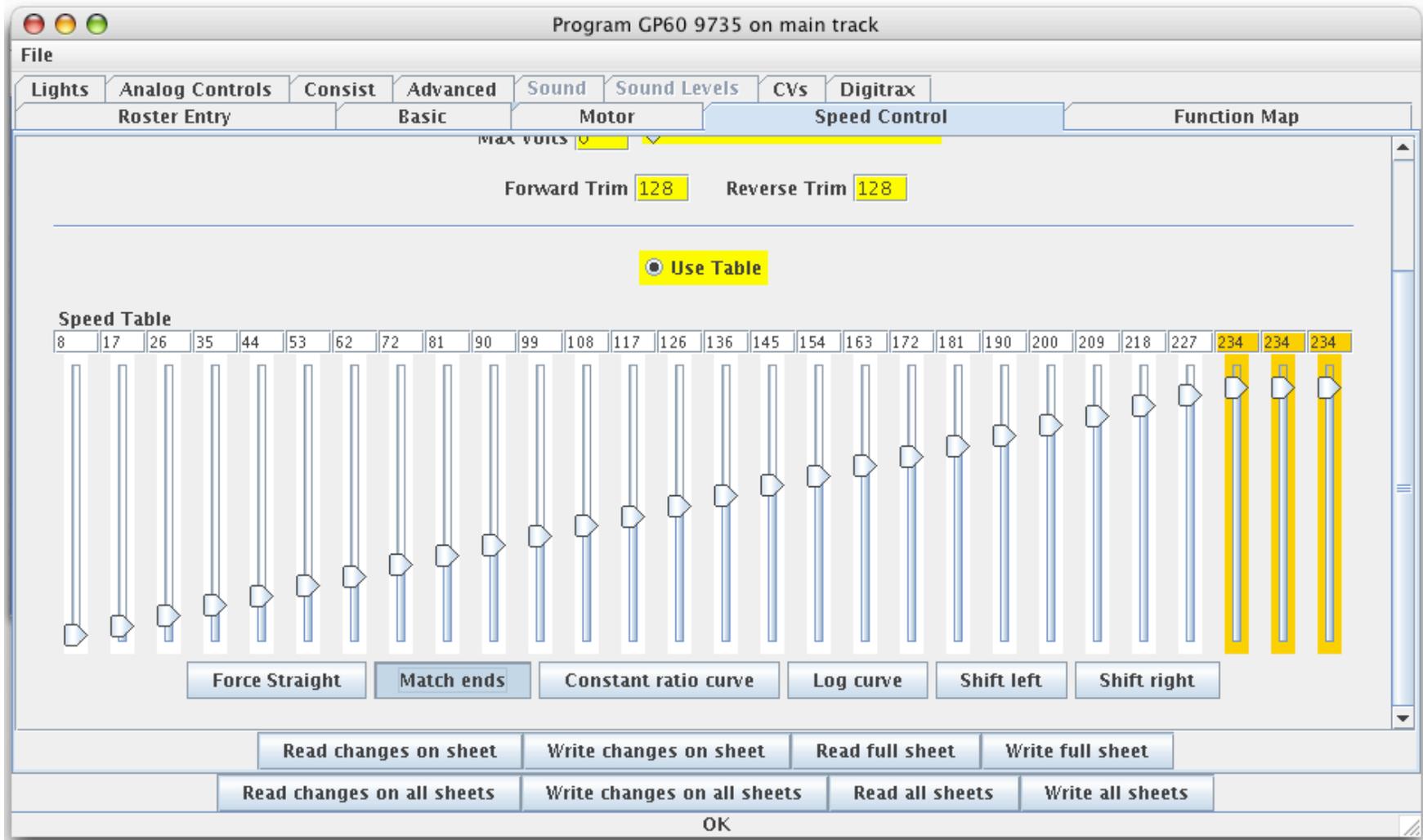
The screenshot shows the Decoder Pro software interface for editing locomotive data. The window title is "Program GP60 9735 on main track". The "Speed Control" pane is selected, showing various fields for the locomotive's configuration. The fields are as follows:

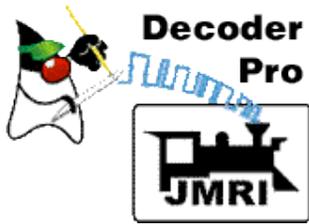
Field	Value
ID:	GP60 9735
Road Name:	SP 9735
Road Number:	9735
Manufacturer:	Lifelike Proto 2000
Owner:	Dave Duchamp
Model:	GP60
DCC Address:	9735 Long
Comment:	
Decoder Family:	Series 3 with FX3, silent, readback
Decoder Model:	DH163L0
Decoder Comment:	Number board light hooked to F1
Filename:	GP60_9735.xml

Buttons at the bottom of the pane include "Save" and "Reset to defaults". The status bar at the bottom of the window shows "idle".



Select “Use Table”, and reduce the value in the 28th (last) step of the speed table.
Click “Match Ends” to adjust all steps.





Click “Write changes on sheet” to send the new table to the decoder.

The screenshot shows the Decoder Pro software interface for a GP60 9735 locomotive. The "Speed Control" tab is active, displaying a "Speed Table" with 24 columns representing different speed settings. Each column has a vertical slider and a numerical value above it. Below the sliders are buttons for "Force Straight", "Match ends", "Constant ratio curve", "Log curve", "Shift left", and "Shift right". At the bottom, there are buttons for "Read changes on sheet", "Write changes on sheet", "Read full sheet", and "Write full sheet", along with "OK".

Program GP60 9735 on main track

File

Lights Analog Controls Consist Advanced Sound Sound Levels CVs Digitrax

Roster Entry Basic Motor Speed Control Function Map

max volts 0

Forward Trim 128 Reverse Trim 128

Use Table

Speed Table

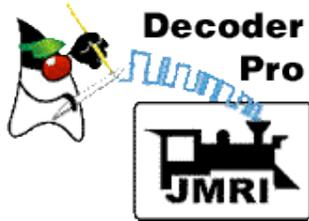
8	16	24	33	41	49	58	66	74	83	91	100	108	116	125	133	141	150	158	167	175	183	192	200	208	217	225	234

Force Straight Match ends Constant ratio curve Log curve Shift left Shift right

Read changes on sheet Write changes on sheet Read full sheet Write full sheet

Read changes on all sheets Write changes on all sheets Read all sheets Write all sheets

OK



As each step is written, DecoderPro indicates progress as shown below.

Program GP60 9735 on main track

File

Lights Analog Controls Consist Advanced Sound Sound Levels CVs Digitrax

Roster Entry Basic Motor Speed Control Function Map

Max Volts 0

Forward Trim 128 Reverse Trim 128

Use Table

Speed Table

8	15	23	31	38	46	54	61	69	77	84	92	100	107	115	123	130	138	146	153	161	169	176	184	192	199	207	215
---	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Force Straight Match ends Constant ratio curve Log curve Shift left Shift right

Read changes on sheet Stop Write changes on sheet Read full sheet Write full sheet

Read changes on all sheets Write changes on all sheets Write highlighted values on this sheet to decoder

Writing CV82...



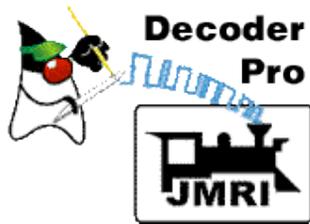
6. Continue to adjust speed table until loco runs the same speed as the lead engine.

Run consist the same speed each trial (50% works well).

Stop consist between speed table adjustments if the speeds are much different.

Note: When programming on the main, CV's usually cannot be read--only written!

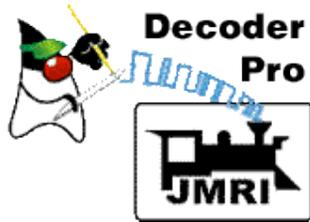
Remember to Save the roster file when done.



Decoder was programmed to match the speed of the two engines, and engines were consisted **without having to know anything about individual CV's.**



Isn't that a great way to program DCC decoders!



Sound Decoder Demo

DecoderPro Animated Demos:

Peter Ulvestad (Edmonton Model Railroad Association)

<http://www3.telus.net/public/ulvestad/DecoderProDemos.html>

