

Model Railway - Direct WiFi Control (DWiC)

System Description and Specification

As developed by the DWiC Working Group

This copy: Version 1.1
150317

1. Executive overview

- 1.1. Interested model railroaders from around the world have formed a working group to design and develop a new concept in Model Railway control called *Direct WiFi Control* or *DWiC*.
- 1.2. The basic concept of DWiC involves the direct control of locomotives and layout using WiFi technology. WiFi is well established and WiFi devices are cheaply available in the consumer market.
- 1.3. The method of control is via a web page interface. The page is stored within the item controlled and so can be customised for that item.
- 1.4. This web page is loaded and displayed by the operator application which in turn can modify the locomotive or layout controllers operation by pressing buttons on that web page.
- 1.5. The system will be able to run on existing layouts that are DC or DCC powered.
- 1.6. The DWiC Working Group is an independent body not attached or committed to any organisation or manufacturer although its members may be manufacturers or belong to other model rail organisations throughout the world.
- 1.7. Any output of the DWiC Working Group will be placed in the public domain for all to use.
- 1.8. The group web page is at:
<http://members.iinet.net.au/~backway/DWiC/DWiCintroduction.html>

2. Contents

- 3. Introduction to the concept**
- 4. DWiC System overview**
- 5. DWiC Components**
 - 5.1. The DWiC Controller**
 - 5.1.1. Mobile Controller**
 - 5.1.2. Fixed Controller**
 - 5.1.3. Hardware Interface**
 - 5.2. The DWiC Operator**
 - 5.2.1. The DWiC Device**
 - 5.2.2. The DWiC App**
 - 5.3. The WiFi Link**
- 6. DWiC System Operation**
 - 6.1. Initialisation**
 - 6.1.1. DWiC Controller initialisation**
 - 6.1.2. DWiC Operator App initialisation**
 - 6.2. Connecting App to Controller**
 - 6.3. Disconnection/handover of controller between Operators**
- 7. Glossary of Terms**

3. Introduction to the DWiC concept

3.1. Background

DCC, the current most popular method of model railway control, was first developed in 1978 by Bernard Lenz a German whitegoods' control unit manufacturer. Bernard, also a model railroader, saw an opportunity to utilize the new microcontrollers he used in his business within his hobby.

DCC had already been commercialised by Marklin by 1990. At this time the DCC Working Group initiated by Stan Ames on rec.models.railroad began investigating existing command control systems. Lenz DCC was soon found to be the better of the systems available at the time. In 1996 the first DCC Standards and Recommended Practices were passed by the NMRA board.

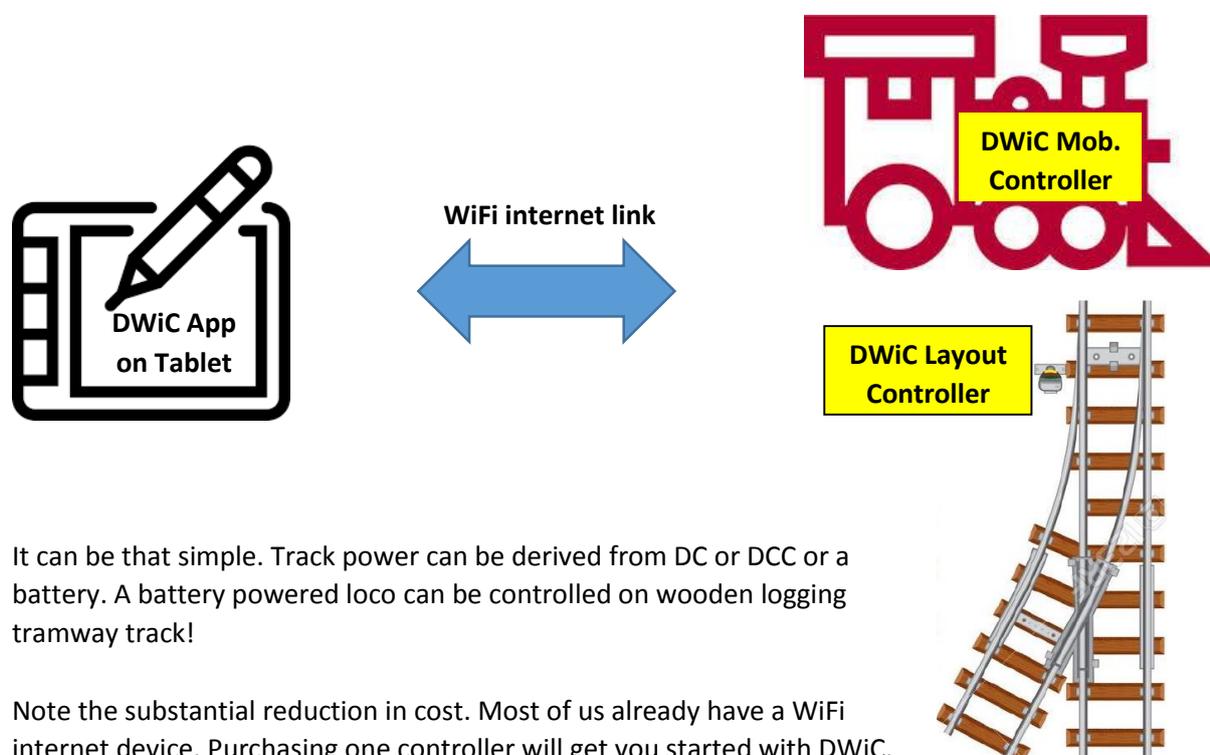
- 3.2. Since 1996 more DCC standards and recommended practices have been certified and DCC technology has grown in what some see as a series of challenging attempts at enhancing an old technology. For example the effort in establishing a return channel – bidirectional control. Often these enhancements have resulted in different manufacturers going different ways and the standards having to accommodate this.
- 3.3. The members of this working group feel that DWiC provides a fresh start to model railway control. This new concept uses well established technology that is currently applied to many consumer, communication, industrial, agricultural and scientific endeavours. The key feature is “we are not alone”, the whole world – outside the model rail hobby is working on this concept. It is time to tag along.
- 3.4. The availability of cheap (US\$3.50) WiFi modules the size of postage stamps has now made the development of the DWiC concept for model railways a relatively simple task.
- 3.5. A key feature of DWiC is that it is neither model rail proprietary nor model rail specific. Once a manufacturer starts producing these controllers (that is adding power supply and output/input hardware to the chips) they could easily be marketed for garden watering, home security, farm automation etc.

3.6. The DWiC Working Group has few boundaries:

1. Must use WiFi as most consumer screen based items have it (and the technology can handle web pages, audio and video in its natural state)
2. The on-board or layout controller must be able to present a web page that controls the item. (otherwise it cannot be "Direct control")
3. An app must be on the operator's phone/tablet/etc that can setup connections, read web pages and control an item via the web page presented by the controller.
4. The preferred form of automated layout control is by using text commands via a controller's web page.

4. DWiC – Basic System Overview

4.1. One of the key features of DWiC is its physical simplicity:



It can be that simple. Track power can be derived from DC or DCC or a battery. A battery powered loco can be controlled on wooden logging tramway track!

Note the substantial reduction in cost. Most of us already have a WiFi internet device. Purchasing one controller will get you started with DWiC.

The tablet in the above diagram is any WiFi enabled device from smart phone to PC. An Android tablet can be purchased for \$A75! Beats any DCC hand controller by far and you can still use it for email and Facebook 😊

Any complexity in DWiC lies in its well established WiFi protocol along with other web standards such as HTML, JavaScript and jQuery – all of which appear invisible and seamless to the operator.

The DWiC controller allows a manufacture or experienced user significant creativity as the web page stored within the controller **is** the human interface. The web page determines what variables are available for that controller and can be customised for each item without compromising the system.

The DWiC App installed on the tablet performs housekeeping such as network and item connection. It then displays the web pages loaded from the selected controllers. The controllers are directly controlled via the web page.

*****The WiFi link** requires further explanation. Two configurations have been proposed and these will be discussed in detail later.

5. DWiC Components

5.1. The DWiC Controller

Various types of DWiC Controllers have been proposed. The common specification for all controllers is that they must be able to initialise and connect to a WiFi network. Then when requested present a web page. Unlike DCC a manufacturer can “do what they please” inside their controller as long as they can store and present a web page when requested.

Some item types identified at this stage are:

5.1.1. Mobile Controller

Not unlike a DCC decoder except that the connection is via the WiFi capabilities of the controller rather than track. Note that WiFi is bi-directional by design and runs much faster (~1Mbits/S) than a DCC signal (~4.5Kbits/S).

5.1.2. Layout Controller

Should be more useable than DCC Accessory Controllers as there is no complicated protocol required to drive them. The only limit to the number of items controlled is the number of buttons/indicators that can fit on the web page and the number of hardware inputs/outputs of the controller. As with mobile controllers, a web page with direct control provides the human interface. The great advantage is that the web page can graphically display the part of the layout it controls.

5.1.3. Controller Hardware Interface

With the exception of the WiFi and track interfaces the hardware of a DWiC Controller can be identical to that of a DCC decoder. DCC hardware interface techniques can simply be rolled over into a DWiC hardware interface. This would include “keep alive” provisions and back emf monitoring. Other interfaces such as “speedometer” can also be added if not already existing in DCC.

5.2. The DWiC Operator

The DWiC operator is the human operating the DWiC operator device.

5.2.1. The Operator Device

The Operator Device can be any WiFi equipped device that can run a DWiC operator App and thus display a controller's web page. Examples are smart phone, tablet or PC. Except for the need to design an App for a particular operating system the DWiC should appear as platform independent.

5.2.2. The Operator App

A conceptual operator App is at:

<http://members.iinet.net.au/~backway/DWiC/DWiC%20conceptual%20App.html>

Initialisation:

- The App establishes connection with the layout network and compiles pick lists of available items separated into item types (eg mobile, layout, services, dispatcher).
- It is expected that the list will be a readable description of the item. For example: "Garraat G42".
- The list should also indicate if the item is "owned", "monitored" or "idle".
- The operator can choose which locos, accessories etc he wishes to own or monitor.

Operation:

- The chosen items are called and their web pages loaded into the App.
- The items directly respond to actions by the operator on that web page.

Handover:

- The App must be capable of handing over an item to another operator, or changing its status (owned, monitored, idle).

Setup

- A list of parameters applicable to that operator and operator device should be user configurable via the App. It can appear to be a separate page/tab.
- Possible parameters should include "Operator Name". This could be stored in the controller as its "owner name".

5.3. Virtual Items

- A virtual item is a WiFi controller with only firmware and no hardware. An example could be a “calamity controller” which in typical PC fashion detects when the operator is under stress and throws in a bushfire or flood.
- Let your imagination run wild.

5.4. Service Items

- The conceptual App shows a typical service item – a fastclock.
<http://members.iinet.net.au/~backway/DWiC/DWiC%20conceptual%20App.html>
- Other items may be timetables, waybills.

5.5. The DWiC Dispatcher App

The dispatcher App would be run on a PC at the dispatcher station. All the typical dispatcher functions could be run by this App. Similar software is already in existence. A tab on the operator App could display dispatcher instructions and communication options.

- Most WiFi devices could easily handle pseudo radio communications to and from the dispatcher.



5.6. The WiFi Network

- This is the component that will require most of the WG effort.
- It will also be the area requiring the most specification.
- The WG have already began discussion on this section.
- Established procedures and protocols must be used
- Decisions need to be made on alternatives. For example two types of network configuration are possible:
 - A star configuration whereby all communication is via a WiFi router. This is probably your home network configuration
 - A Direct WiFi configuration where each item can communicate directly with other items.
- An advantage the WG has which has already been mentioned, is that many others, including outside the model rail world are working on the same thing.

6. DWiC System Operation In Depth

6.1. Initialisation

6.1.1. DWiC Controller initialisation

6.1.2. DWiC Operator App initialisation

6.2. Connecting App to Controller

6.3. Types of connection – control or observe

6.4. Disconnection/handover of controller between Operators

6.5. Automated Control

- A DWiC Controller can have an automation option.
- It has been proposed that automation (aka computer layout control) can be performed by passing parameters as part of the controllers URL.
- It has been further proposed that these parameters be defined as plain text words.
- A suggestion was that a list of “reserved words” and perhaps other permanent information such as loco class and number be included in the web page as metadata fields. Thus an App will know what it can control.

Mathew Geier proposed automated control and controller-App connection as follows...

“Use http to actually send messages to the devices and the message would be something like

<http://04DCE4locomotive.local/control?speed=4&direction=forward?whistle=2>

(Start moving forward, accelerate to notch 4, blow horn 2s) or even relative

<http://04DCE4locomotive.local/control?speed=+2> (which would just increase the speed by two 'notches' over what was previously sent).

In response to these commands the locomotive would just send standard http responses back, with say 'OK' for a message that was accepted and something like 'ERROR. Command not accepted: feature not implemented'. A program sending the messages would just look at the http protocol error codes and decide what to do. A human directly controlling the locomotive would get an error they can at least understand.

A point message would be like

<http://05345turn.local/control?direction=right>

These messages might look to very verbose and take up lots of 'air time', but really even at WiFi's minimum 1Mb/s speed, these are going to be on the 'air' for very short periods of time.

Note using http to send control messages is exactly what the SIP protocol used by Voice Over IP telephones / video conference machines does. It replaces the older complicated 'binary' H323 protocol. We should advance along similar lines and also co-op http as the transport to actually send

control messages like the SIP committee did. (Although they don't use the same 'port' as http, they got a new 'SIP' port number defined for their use, but SIP is sufficiently similar at a low level to http that people don't need to implement it from scratch.)

The names of the devices (04DCE4locomotive.local) would be discovered by some application running on a 'user interface' controller (or some sort of master controller for larger layouts) at power on by using the mDNS protocols. (Apple calls mDNS 'bonjour', it's quite important on Macintosh networks, less so on Windows based ones, but it's certainly a mature and well used protocol).

A locomotive added to the layout 'mid session' would just power up, go through WiFi association, Zeroconf and start broadcasting it's name by mDNS. The users 'controller' could potentially then pop up a message 'new locomotive discovered' and lead the user through what is required to 'integrate' it into their operations. They could just ignore it too - and the locomotive would just sit there where it was placed doing nothing until some one else 'picked it up'.

There is also no reason why the address of a locomotive (the default potentially would be based on the wireless chip's ethernet MAC address) couldn't be a configuration option stored in the Micro's nonvolatile RAM, so the mDNS for a locomotive could potentially be it's road number e.g. after initial configuration it's 'address' is up4014locomotive.local. and not 04DCE4locomotive.local (Note mDNS already has ways to deal with name conflicts, so if two devices with the same name appear, that case can be handled.)

There may be a need to define some sort of 'what I support' message, where the 'device' (be it a locomotive or layout controller) can be asked to report it's capabilities. This response message has to be in some sort of structured text so it can be parsed by a computer program but also can still be understood by a human. XML is often used for this sort of thing, but is probably overkill. But the text does have to be structured in such a way that it can be parsed by a computer program and that extensions don't 'break' older parsers.

The only 'fly the ointment' in what I propose is there probably needs to be a way to broadcast an emergency 'all stop' command. That probably will have to be a 'listener' on each device listening for a particular pattern (binary bits) broadcast on a particular port. http doesn't lend itself to broadcasts.

One advantage of using http in this way is we have an infinity extensible control protocol. We need a standardised core but there is opportunity to allow 'vendor' messages that could do things like upload custom sound bytes on the fly, so while the 'blow horn' command probably should be a standard command, a 'vendor' extension could allow uploading for new sounds on the fly while the locomotive is actually operating.

And this ties in with the embedded control UI that Bob mentions in his 'design brief'. The buttons on the simple embedded 'home' page in the locomotive would just be links that generate 'standard' control commands.

7.0. Glossary of Terms

Term	Description	Reference
Direct WiFi Control (DWiC)	A system of controlling model railway <i>items</i> directly via a WiFi interface. DWiC is pronounced as Dee-why-see or more affectionately as “dwick”.	
DWiC App	A software application that runs on a WiFi equipped consumer device such as smart phone, tablet or PC.	
DWiC Operator	The human who is operating a DWiC controller via an App	
DWiC Controller	A microprocessor device having the firmware and additional hardware necessary to interface via html to the App(s) and control an item such as a loco or layout.	
Item	Anything controlled by a DWiC App	
Mobile controller	As is found in a locomotive	
Layout Controller	As is used to control fixed devices on a layout. For example turnouts.	
Star WiFi Network	A network whereby all communication is via one node known as a router	
Direct WiFi	A network where each WiFi device can communicate with every other device.	