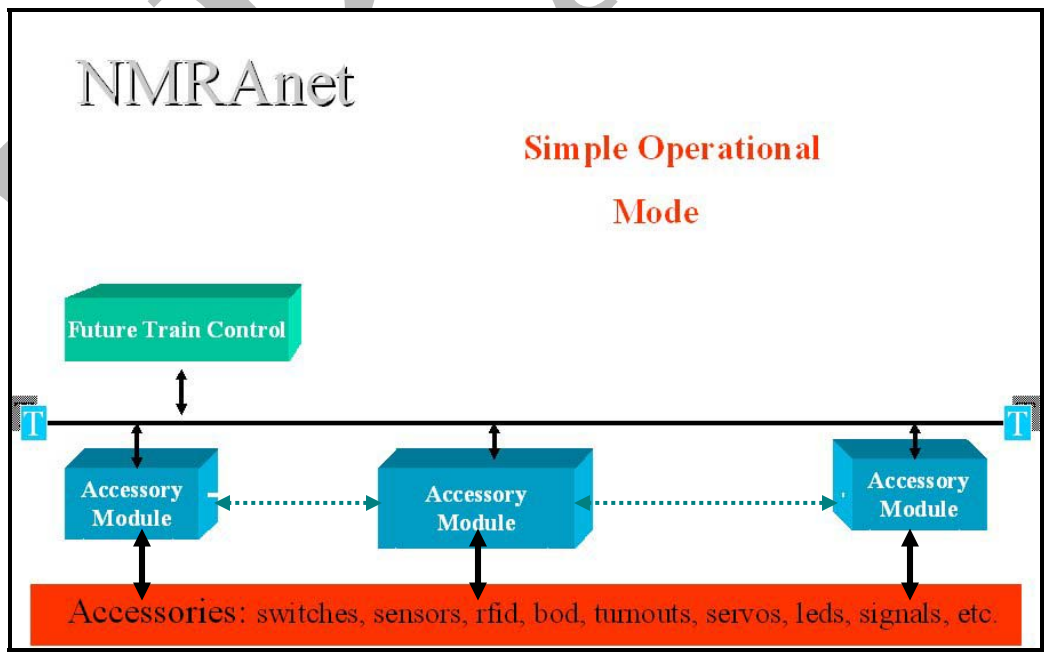
This document provides an overview of the NMRANET™ control bus that is intended to control the operation of accessories and trains on a model layout. Sub-standards describe the technical aspects of this bus.

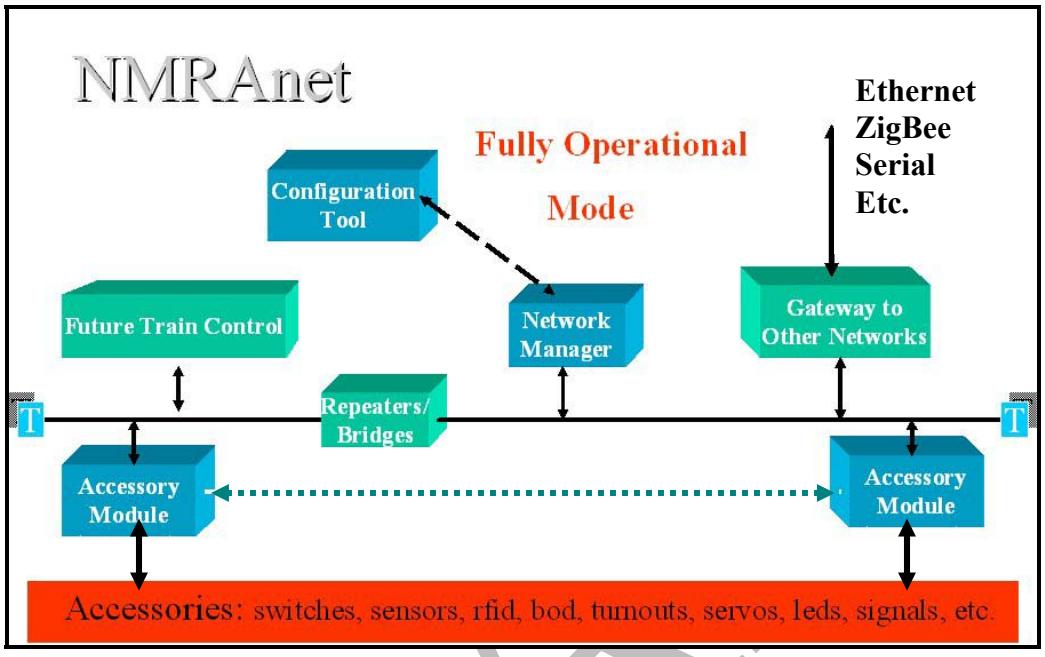
NMRAnet is a flexible and robust means of connecting multiple accessories to each other in a peer to peer relationship. The primary network technology is CAN (Controller Area Network) which has been used in cars for the last 15 years and has been expanded into industrial automation. The network is designed to be able to be expanded into other network technology (Ethernet, ZigBee, etc). The CAN technology enables NMRAnet to be based on a known and proven technology/standard with many standard micro controllers designed with an internal CAN module. With CAN's maturity accessories modules can be developed at a much reduced cost.

NMRAnet can support three modes of operation: 1) **Simple Operational Mode**, with a limited number of modules where simple accessory relationships can be established without the use of a computer; and 2) **Fully Operational Mode,** where the network can grow to almost any size with additional user tools available to assist the user to establish and maintain a robust network; 3) **Extended Operational Mode**, where several layouts can be brought together in a large club or meet layout.

In NMRAnet, all modules shall support the Fully Operational Mode, but may also support the Simple Operational Mode. This allows the user to grow his network and at some time upgrade the network without loosing his existing modules and be able to fit into larger club layouts.



In the **Simple Operational Mode**, modules can be connected to the bus, and with the pushing of buttons and switches on each module, the events can be communicated between modules to activate devices. No Configuration Tool or Network Manager is necessary. However, the operator must be fully knowledgeable on the operation of the bus in the event errors occur.



For the **Fully Operational Mode** configuration, the number of accessory modules and their functions can quickly overwhelm the knowledge of the user. As a result, a Configuration Tool and a Network Manager become necessary. The Configuration Tool can be disconnected once the devices are working. The Network Manager always stays connected and monitors the health of the bus. The Network Manager does not get involved in operating the bus and the many events that occur; it only monitors the events to make sure the bus stays healthy.

NMRAnet has the following distinguishing features in the Full Operational Mode:

1. There is a Network Management function which provides automatic power up sequencing, automatic module addressing, node guarding for error detection, and network status and debug capability.
2. Each module has its own description of capability and configuration carried in a compressed XML file. This allows a Configuration Tool to treat each module as a separate device.
3. Almost unlimited expansion to 65,000 modules spread over multiple networks.
4. Can handle up to about 2000 messages per second.

The Producer/Consumer Model of establishing Accessory Operation

Producers are of many types; sensors, switches, block detectors, A/D converters, (unlimited). They are open, closed, on, off, set, detect, unoccupied, occupied, (unlimited).

Consumers are of many types; turnouts, LEDs, lights, signals, semaphores, motors, (unlimited). They are throw, thru, on, off, clear, stop, rule 286, stop and proceed, proceed, locked, unlocked, on then locked, unlocked then off, (unlimited).

The Producer / Consumer model allows many-to-many connections easily. Second, and perhaps more importantly, it allows connecting devices without any standards for data interchange format. In other words, you can connect very different types of devices that in the source/destination model might not be compatible. This **interoperability is something not to be underestimated in networks** with many devices interconnected, as is common on a model railroad.

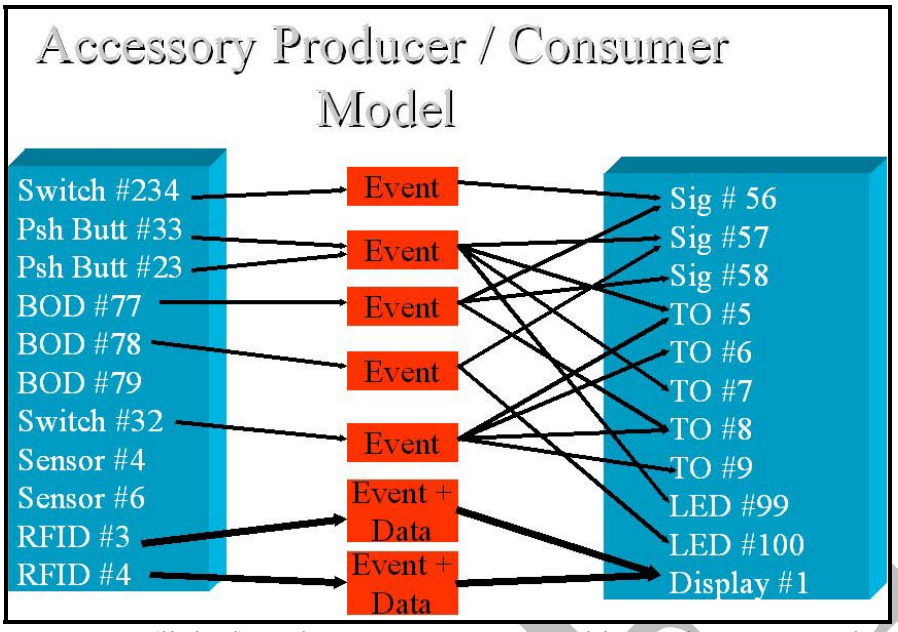
Messages are produced (broadcast) by devices and then consumed by other devices. These messages contain an identifier followed by optional data. There is no information whatsoever about either the source or destination of these messages. Likewise, in many cases the message consists of the identifier and **no data**.

Devices in this scheme are all loosely coupled. Producers create and send messages without knowing anything about the devices that will consume those messages. Likewise, consumers receive and process messages without knowing which device sent the message.

The user and the manufacture of these devices are free to choose any and all with whatever combination of capability is needed to either sell the product or to accomplish the needed control on the users' layout. Producers are independent of other Producers. Consumers are independent of other Consumers. Producers and Consumers are independent from each other.

NMRANet needs one method to tie or link the many types of Producers to the many types of Consumers. This is done in a Peer-to-Peer broadcast to allow all possible combinations to accomplish the many and varied relationships:

1. One-to-One
2. One-to-Many
3. Many-to-One
4. Many-to-Many
5. Producers-to-Computer(state machine)-to-Consumers



Here the **Event** gets **tied** (linked) to the **Producer(s)** to enable Producers to send out (produce) an Event number. Each type of stimulus or state of the Producer is able to be tied to an Event. The **Event** also gets **tied** (linked) to the **Consumer(s)** and, with this tie, an **action** is specified, telling the Consumer what to do when receiving an Event.

The Event is not a Command, it is not a Device-ID, it is not an ID plus command. The **Event** is the **Tie** or **Link** between Producers of different types and capabilities and Consumers of different types and capabilities. The Event cannot be a command by its definition and usage. After an Event# is tied to both Producers and Consumers, the user if he wishes can assign an English description; Route 66, etc.

The other aspect of the Accessory P/C model is that an Event can carry **optional data**. This is an **EXTENSION to the P/C mod**el. This aspect however has restrictions and limitations. There is **no longer independence** between a producer and the consumer of the optional data. Optional data must have a common format and definition between the producer and the consumer.

Data should only be used with Events with great care, as they tend to undo the good features of Events, namely state-independence. For example, it is preferable to use several different Events for a device that only has a few states. Even devices with many states, such as an analogue converter might be better served by defining a lesser number of Events such as a number of thresholds, rather that the raw data.

Extended Operational Mode