Commissioning a wireless sensor and control system

By Jason Choong and Zachary Smith

ZigBee sensors are quite easy to install individually, but correctly designing and installing a large network is another matter. Jason and Zachary outline three of the most significant challenges in a big sensor network and describe a variety of solutions that can help overcome them.

Embedded wireless sensor and control systems such as ZigBee provide developers with the infrastructure to create wireless versions of existing sensor and control applications. These technologies can reduce or eliminate much of the cost of wiring, provide greater flexibility with device placement, and enable more scalable systems with meshed networking.

However, these benefits come with additional challenges in commissioning and deploying. The following discussion introduces three of these significant challenges and presents a variety of solutions to address them effectively.

Deployment and commissioning challenges

One such challenge is the *identification* problem. Where sensors or devices are wired to other devices, they can be physically traced to each other. However, this is not possible with wireless. In many scenarios, multiple devices of the same type may be turned on at roughly the same time, so how can one identify a particular device?

In a building with 100 identical sensors (such as temperature sensors), a tool can detect all 100 devices, but how does the installer identify which of the 100 devices is the one in room 16A, given that all devices are functionally identical? How does one identify and select that sensor, place it on a particular wireless network, and make it affect the behavior of the controller in section 16 of the building?

In its general form, this challenge is about mapping logical devices to physical devices, as shown in Figure 1.

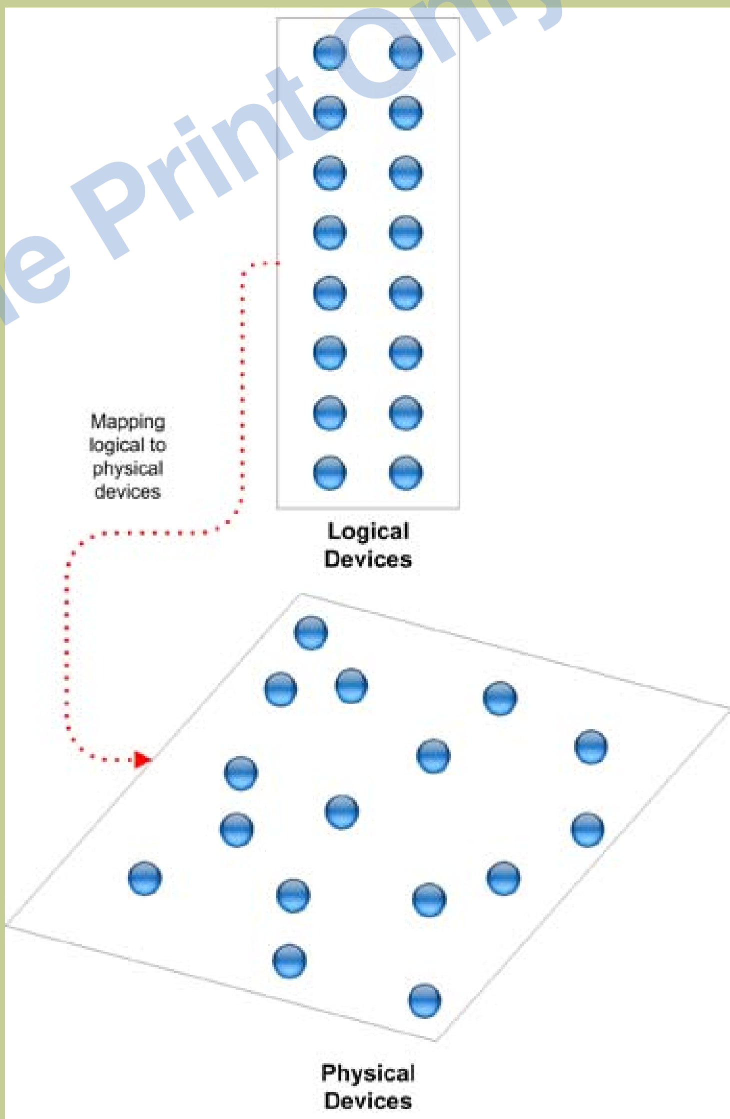


Figure 1

Another challenge is the *network design* problem. How does one design a network that can minimize interference from other wireless technologies? Which channel should be used? Should devices be placed on the same wireless network or segmented into different networks? This problem also encompasses considerations such as the number of wireless routers to use, the ratio of routers to end devices, and device placement.

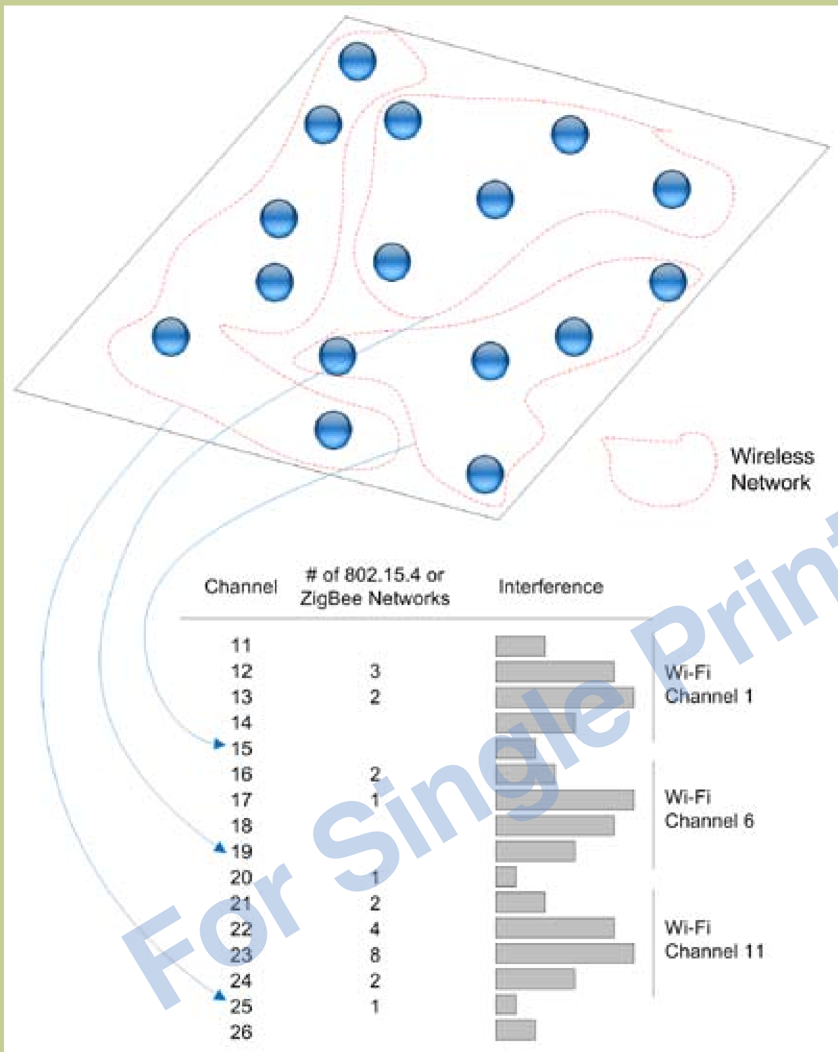


Figure 2

motion being sensed on specific sensors? Bindings provide connections between devices. These connections replace the physical wiring by setting up logical control paths between devices. An example of binding relationships within a network is shown in Figure 3.

Solutions for these problems must consider certain practical realities. These include the installer's skills and the availability of tools (or otherwise). In most environments, the end user or installer has no knowledge or understanding of wireless technologies. In others, the installer may have little or no access to tools. Solutions must fit these constraints.

Solving the identification problem

A variety of solutions can solve the identification problem. The main task of these solutions is to provide a means to map logical to physical devices. The need to define a mapping largely results from having identical devices on a network in the absence of distinguishing features.

Sequential activation or commissioning

The simplest solution is to avoid having a collection of logical devices that requires mapping to physical devices in the first place. If the installer has the means to turn on individual devices one at a time, and then take appropriate action on each device, he or she has effectively identified each device. Another way of doing this is to have a button on each device and,

An example is provided in Figure 2. Multiple networks are to be deployed in an environment with existing interfering sources (such as Wi-Fi routers) or other ZigBee networks. The challenge is to establish which channels and networks to set up and then place devices in effective locations.

The third challenge is the *binding* problem. Devices on the same network may communicate to any other device, but which devices should actually be communicating to each other? For instance, as part of solving the network design problem, 12 motion sensors might be assigned to the same network, but which lights should each of these 12 motion sensors be bound to in order for the lights to be automatically turned on as a result of

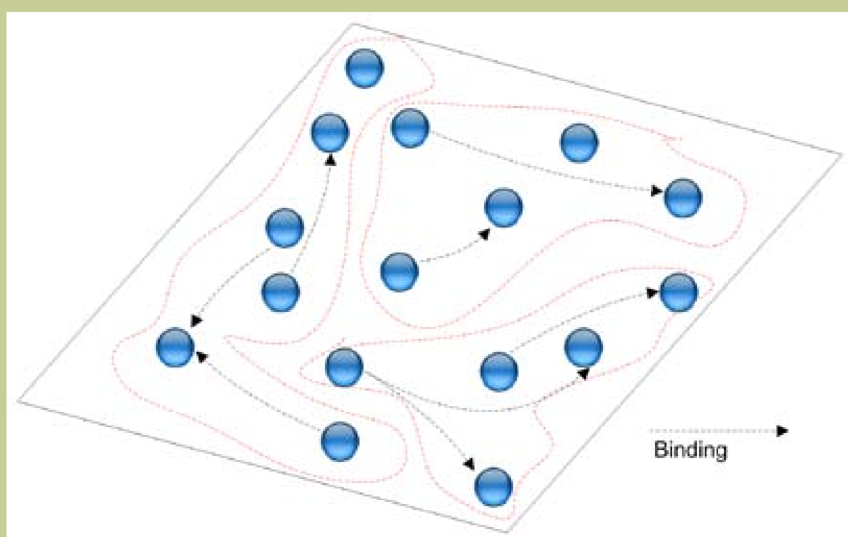


Figure 3

through the commissioning process, require that this button be pressed to effect commissioning only on this device.

Identify thyself

In scenarios where physical access to individual devices is not possible or sequential commissioning is too slow, alternative methods are required. ZigBee provides an Identify Command the device could use to provide an audible or visual feedback. Devices are turned on, and a tool could show a list of detected devices. The installer then selects a device on the tool to generate an Identify Command. This stimulates an audible or visual feedback, thereby identifying the device. The installer can then decide to take action on that specific device if it is the correct one. This method only works when some physical feedback mechanism is available and accessible to the installer.

Automatic locationing

Another solution is to find some way of automatically locating devices on a plan. Using automatic locationing methods (GPS, ZigBee-based, or some other method), devices could be automatically placed on a blueprint or floor plan. The installer can then take action on devices based on where they appear on the plan. The effectiveness of this method is dependent on the accuracy of the locationing algorithm relative to device density.

Out-of-band methods

A wide range of methods falls into this category. Fundamentally, they involve some means of associating a device identifier to a physical location on a plan or some physical identifier (such as a room or floor number). While similar to automatic locationing, this method differs in that automatic locationing may be somewhat less accurate but is more convenient, as it requires minimal or no additional work to identify the device.

Out-of-band methods involve some custom or special methods to tag devices outside (out-of-band) of the usual ZigBee or wireless communications.

An example is provided in Figure 4. Devices can be manufactured with a barcode, which is associated with a device ID (such as an IEEE address for ZigBee). The barcode is then scanned, and the installer associates that barcode with a position on a (paper or electronic) plan or perhaps a room identifier. These then allow for later identification when an action needs to be taken on a specific device by using the physical location or a room identifier.

Solving the network design problem

Network design is affected by the operating environment and the requirements of the application. Environmental factors include other radio networks, radio propagation, and available locations to place devices and routers. Application characteristics that impact network design include device density, how often devices communicate or sleep, and sensitivity to performance.

Channel selection

Channel selection is mostly influenced by the environment. In particular, avoidance of interference from Wi-Fi and other 2.4 GHz-based technologies may impact the channel choice.

ZigBee has two methods for optimizing channel selection. It can be set up to automatically select the channel with the lowest energy (or interference) when the coordinator is first started up. It also has a mechanism to change the network's channel when interference is believed to be causing significant problems.

Additional steps can be made to improve channel selection. These may include taking energy scans across the entire building or operating environment (the coordinator itself may not detect interference in a remote part of a building) or proactively initiating channel changes prior to problems occurring.

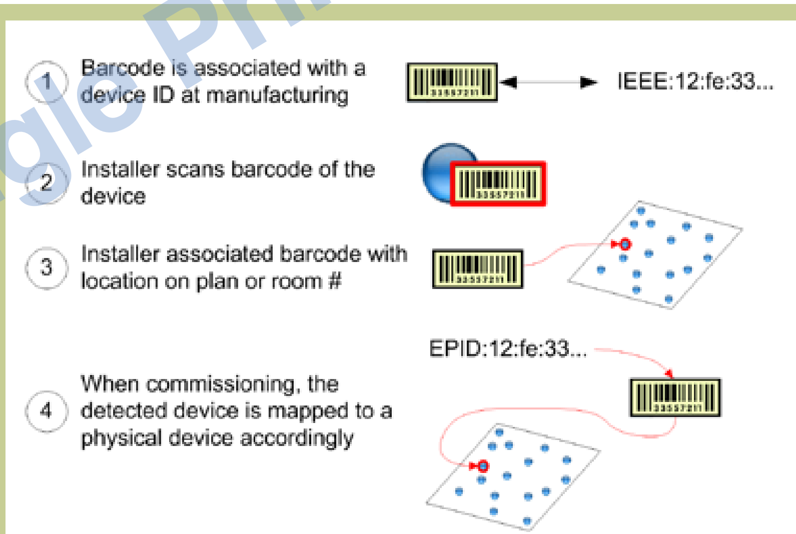


Figure 4

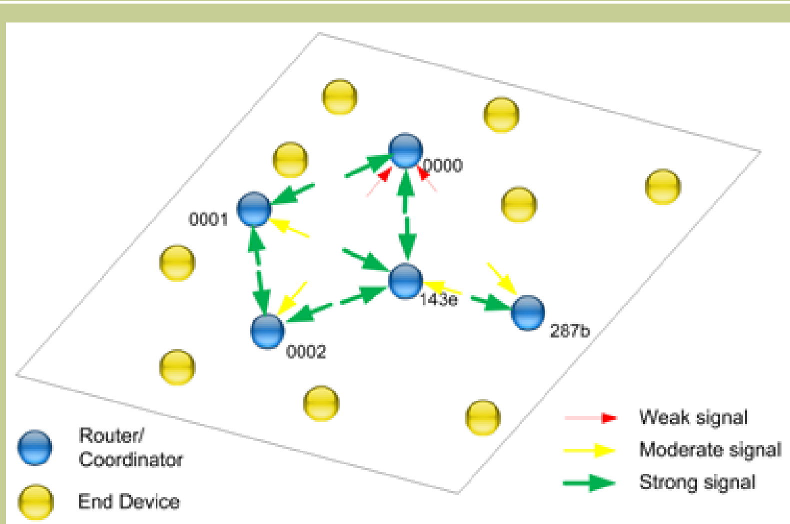


Figure 5

Network planning

Network planning encompasses a wide range of issues. For instance, how many wireless routers are required, or more aptly, what is the ideal ratio of routers to end devices for a specific application? Where should routers be placed?

The key metric for determining this is connectivity. Wireless mesh networking works best when every device has more than one path to reach the rest of the network. This provides resilience against temporary signal loss from radio fading or more permanent signal loss from a newly introduced obstruction or interfering radio source. It allows devices to find alternative routes to reach other devices.

Pre-release testing will provide some indicators on router-to-end-device ratios. In actual deployments, installers can deploy devices to these ratios and, using reports on connectivity, add additional routers or place them in alternative locations to improve connectivity.

An example is provided in Figure 5. With the exception of router 287b, the routers have good connectivity to at least two other routers (as shown by colored arrows). Should the only good connection available to 287b fail (through 143e), 287b may be isolated from the rest of the network.

Another consideration is the way the ZigBee network should be segmented. Should a single ZigBee network be used to provide the infrastructure for all applications in a building or factory? Should several networks be constructed to provide independence for different applications and systems – one network for lighting and security and a different one for machinery? Alternatively, the networks could be partitioned by location – even floors for one network, odd for the other to reduce device density.

Using a single network reduces the need for duplicate infrastructure. Only one set of wireless routers and a single wireless security system are needed. Conversely, a single network may also result in performance degradation since more devices use the same channel and routing infrastructure.

Exactly what the devices-to-network ratio is and how those networks should be spread across different channels is highly dependent on the application, radio channel utilization, and device density.

Devices also could be on different wireless networks but still be on the same "network." This can minimize interference due to high device density or allow remote wireless networks to be on the same network. By using ZigBee bridges, devices can communicate to any other device across wireless networks on different channels and in more remote locations, as shown in Figure 6.

In this example, devices in building #1 are segmented into two different wireless networks on different channels to minimize interference, while TCP/IP infrastructure is used to bridge these networks together with a third wireless network in a more remote building #2, creating a single ZigBee network for all devices.

Solving the binding problem

By this stage, the means to identify devices has been determined. Devices are now on the networks they need to be on. The next challenge is to determine the control path between devices – that is, how devices should be bound to each other.

Binding is application specific. A light switch and a light fixture should be bound together, but the light fixture should not be bound to a thermostat. However, the switch could be bound to an air conditioning unit to turn it on or off.

The specifics of how devices are bound may be application dependent, but the binding procedure can be generalized into a few common methods.

Binding through stimulus on devices

In applications with simpler binding relationships, such as between a single light switch and fixture or between an air conditioning unit and thermostat, binding decisions are relatively simple. Buttons or Dual In-line Package (DIP) switches could be provided on devices to allow the installer to make binding decisions directly. These often take the form of buttons with timers ("if two devices have their binding

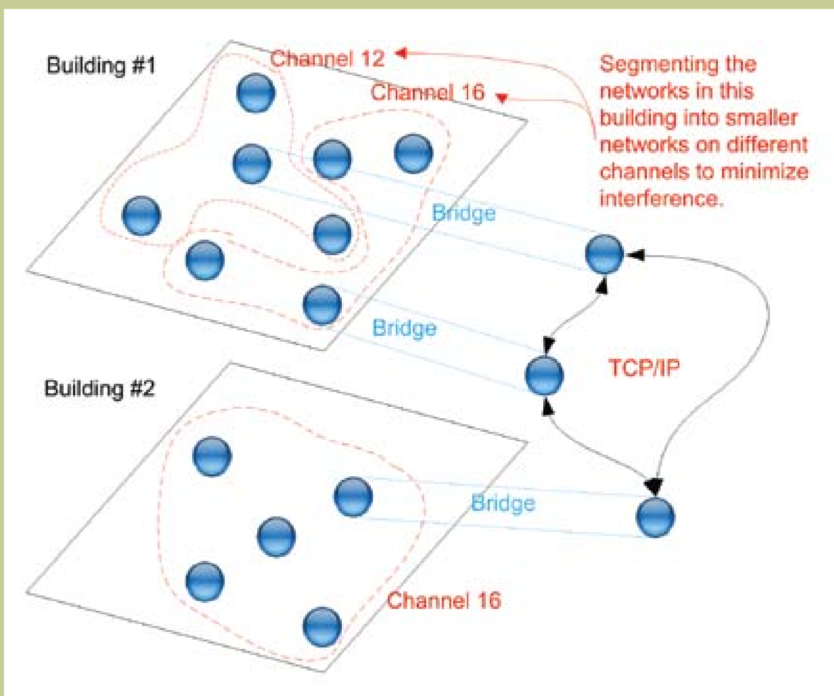


Figure 6

buttons pressed within *x* seconds of each other, they are bound to each other") or switch settings ("two devices that need to be bound to each other should have their DIP switches set to the same value").

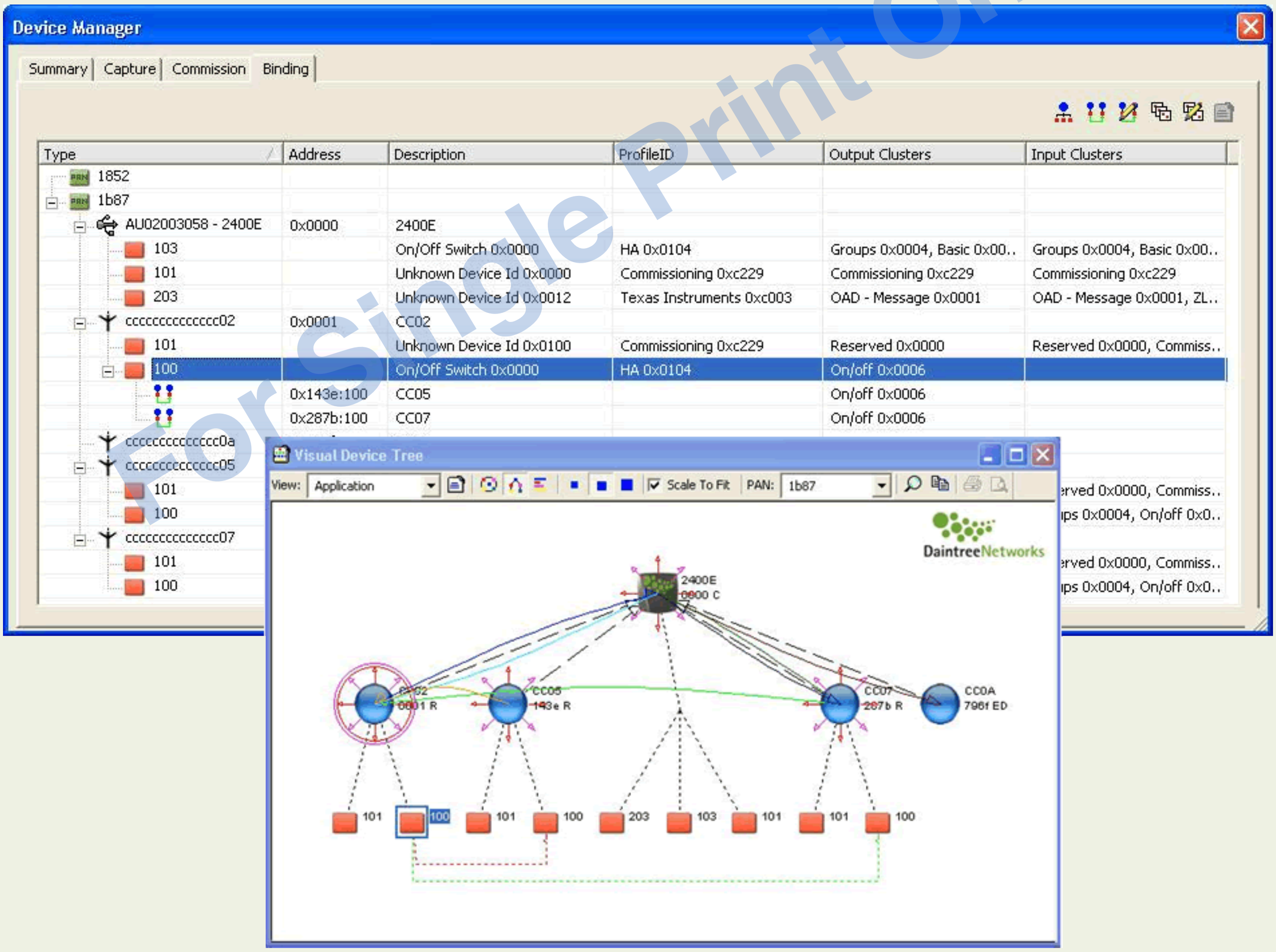
Automatic bindings

The ideal binding method is to have bindings set up automatically. This is viable in certain applications. For example, in an application comprising a large number of sensors and a single gateway that collects all this information, no special binding is required.

Powerful testing and analysis in one solution

Daintree's *Sensor Network Analyzer (SNA)* provides one of the industry's most comprehensive solutions for ZigBee and 802.15.4 testing, analysis, and commissioning. The SNA simplifies the configuration and commissioning of 802.15.4/ZigBee systems during development, field trials, deployment, and maintenance with easy-to-use GUI and standards-based, over-the-air commands to ensure interoperability.

Network visualization, locationing, and API features provide the tools required to design and verify device identification solutions.



Network design parameters can be obtained using network analysis, channel scans, performance analysis, and over-the-air startup commissioning. Through binding and group commissioning features, control paths and bindings can be set up quickly and verified.

The SNA combines powerful testing and analysis tools (ideal for field trials) with an intuitive user interface and unique visualization capabilities (ideal for installers). Incorporating the latest industry standards, this is a beneficial solution for any ZigBee deployment.

All sensors send their information to that single gateway.

More complicated applications also may be automated. For instance, if through the identification stage room numbers are used to identify device location and devices in each room are always bound to each other, devices found to be in the same room could be automatically bound to each other.

Manual or machine-based binding

In more complex applications and installations, other methods are required. For instance, if the behavior of a variable air-volume controller may be determined by multiple temperature sensors, thermostats, and motion sensors, manual or machine-based control systems are required.

This final class of requirements deals with more complex scenarios that cannot be adequately handled by simpler "buttons on devices" or be completely automated with simple algorithms. Here, tools or control systems are required to set up more complex binding relationships.

Work orders and the user interface

Solutions for identification, network design, and bindings may involve complicated algorithms or complex protocol interactions. However, work orders and the user interface for tools used by the installer should be simple. It is expected that most, if not all, installers will have no knowledge of the underlying communication technology, or even wireless for that matter. What is presented to the installer must be a lot simpler.

Consider, for instance, the complexity shown previously in Figure 5. To interpret the measurements shown in this diagram, an understanding of radio communication is required. The algorithms used to determine sufficient connectivity between routers should be hidden from the installer. Summary information should be provided instead, as shown in Figure 7. Green status symbolizes sufficient connectivity, while yellow or red status alerts the installer to insufficient connectivity and is perhaps accompanied by simple recommended actions to resolve the problem.

Choices available to installers

This article describes three key commissioning challenges – identification, network design, and binding – and offers a variety of solutions. The choice of solutions, work orders, and user interface to the installer is highly dependent on the nature and requirements of the application, the installer's skills, and the tools at the installer's disposal. IES

Jason Choong is the director of marketing for Daintree Networks in San Jose, California. Prior to joining Daintree, Jason held roles with Agilent Technologies and Hewlett-Packard. His extensive communications industry expertise includes IP, ATM, and SONET technologies. Jason is also active within the ZigBee Alliance and is the chair for the Commissioning Tools Task Group (CTTG). He has a BS in Engineering and Mathematics from The University of Western Australia and an MS in Engineering from Melbourne University.

__Zachary Smith__ is Daintree's CTO. He serves as the technical editor of the ZigBee specification and the ZigBee network layer specification, the vice chair of the ZigBee architecture task group, a member of several ZigBee committees and working groups, and a participant in IEEE 802.1.5.4b. Prior to joining Daintree, Zachary was CTO of BM Group and chief software architect and founder of Ember Corporation. He has worked as a professional rock musician, and his perhaps most enduring accomplishment is the home-video version of Donkey Kong for ColecoVision. He has a BA in Music from Hampshire College and an MS in Computer Science from the University of Massachusetts.

Daintree Networks, Inc.
408-351-3645
jchoong@daintree.net
zsmith@daintree.net
www.daintree.net

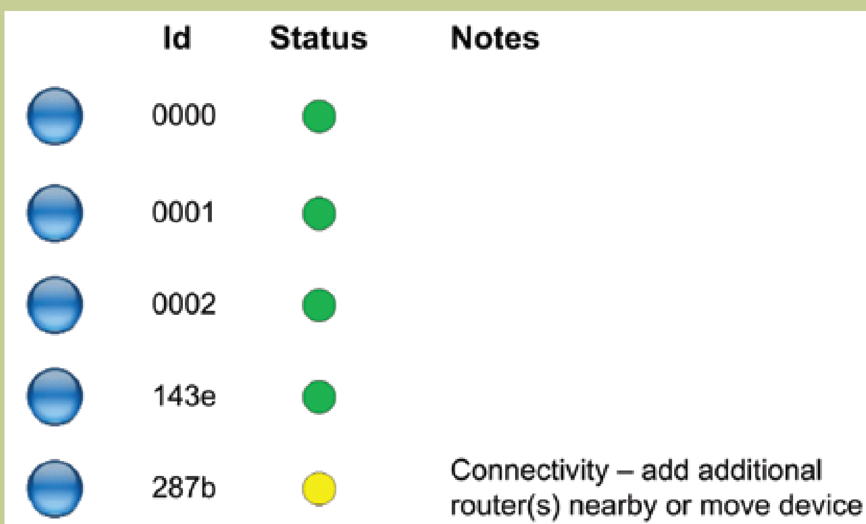


Figure 7