CHAPTER 8

Pathways to the Internet of Things

This book has described the details of an emerging new architecture for the Internet of Things. But new architectures rarely displace legacy systems unless there is an overarching benefit that drives their adoption. For the IoT, the major benefit can be expressed in the unique new relationships possible between the myriad end devices and the big data servers that analyze and control the data flowing to and from those end devices.

Data Drives a Change

Fundamentally, the coming billions of Internet of Things devices will simply generate *too much data* to be analyzed in traditional ways. Instead of the usual one-to-one predefined IP legacy topology, only a publish/subscribe model allows the big data servers to be selective and adaptive in the choice of data to operate upon, and is thus smarter over time.

Even more importantly, the big data analyzers will not even know what data streams would be useful *until they discover the data*. Information neighborhoods created through data stream affinities will present opportunities for selecting and combining small data flows from many different kinds of end devices, not all of which are even part of a specific application. This allows IoT applications to become smarter and smarter over time, as ever more end devices are installed (see Figure 7-4). Whatever initial purpose these end devices serve, they may also unexpectedly and unpredictably benefit other applications that discover their data outputs and find them useful (if the chirp streams are made public).

When initially installed, specific appliances, sensors, and actuators may serve a particular application. But over time, new end devices may be deployed by the same or other organizations. Data streams from these new devices may also be recognized by "affinities" of place, time, or correlation to be incorporated into the original application's information "neighborhood."

Classification is the Challenge, Chirp is the Answer

So if the only way that IoT can reach its potential is through (often) ad hoc publishing and subscription of data streams, what does that say about the data being sent and received by end devices? Simply put, that data must be *externally* classified so that future

known and unknown subscribers can locate, identify, and act upon it. This is completely different from traditional IP networking, in which the external packet components are essentially generic, and thus any classification (moisture sensor versus streetlight versus toaster, and so on) must take place within the data payload itself. In essence, the packet structure of the chirps is *potential knowledge*; chirps are not merely the *containers* of information.

The self-describing classification inherent in the very structure of the chirp packet (refer to Chapter 6) is designed to make publish/subscribe relationships possible across applications, vendors, locations, and time. These self-describing classifications will identify characteristics that allow data subscribers to distinguish between all manner of sensors, actuators, and other devices. This is the prerequisite first step toward determining whether the data being generated by these devices is potentially useful and is necessary to make possible a publish/subscribe network with the eventual scope of the Internet of Things.

The power of self-classified data streams is the fundamental driver of a new emerging IoT architecture. (Even if IP capability in all devices were free, *and it's not*, there would remain a need for a set of commonly understood self-classifications carried within the IP packet payload to enable broad publish/subscribe utility, as shown in Figure 8-1. (See the following "Chirps in IP Packets? Why?" sidebar.) The steps of implementing the network architecture needed to create and transport these self-classified data streams are the subject of this chapter.

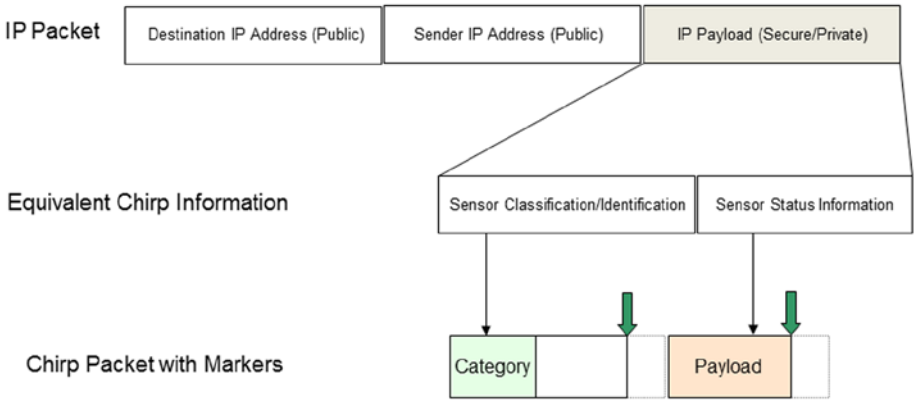

Figure 8-1. *An important distinction between chirp–based IoT packets and traditional IP is that the classification of the data type is part of the public and private markers of the chirp packet—easily "seen" and quickly acted upon by intermediate networking devices. By contrast, the only possible location for self-classification in IP packets is within the payload itself, which requires slower deep examination of the packet at intermediate hops*

The Ends are the Means

This book has described an emerging new architecture for the Internet of Things, designed to address the realities of connecting billions of relatively unsophisticated devices at the very edge of the network. The case has been made for a new terse self-classified protocol of chirps to be the communications medium to these devices, but there are currently no *commercially available* chirp end devices or chirp-enabled propagator nodes. The need for light, purpose-built protocols and devices is revolutionary, and these are early days.

An overnight replacement of existing IP networking protocols in the Internet of Things is impossible—and fortunately will not be required. As with most networking evolutions (twisted-pair Ethernet, Wi-Fi, and so on), the end points will eventually be the major numerical and technical drivers for change, and the support of both chirp and IP protocols to end devices side by side will be necessary to allow for network transformation. This will also be true for existing big data servers at the core of the nascent IoT: they cannot be changed out instantaneously. Fortunately, the propagator node architecture provides an ideal means for a gradual ("and") migration to take place, as described in detail here.

Many different organizations will play a role in the promulgation of the chirp-based Internet of Things. The suppliers of the thousands of types of end devices (from appliances to sensors to automobiles) will work with industry leaders in silicon integration and platform technology such as Intel Corporation to create integrated "chirp chips" in many different configurations and price points. Networking suppliers and home automation developers will build propagator nodes and also incorporate propagator node technology within existing types of equipment such as switches, routers, access points, set-top boxes, and more.

Carriers will make adaptations to the emerging chip-based architectures, many likely offering cloud-based services for interpreting and analyzing the small data flows from chirp streams, perhaps in combination with existing big data system suppliers. Large global Original Equipment Manufacturers (OEMs) will likely also be an important first class of customer and an early promoter of chirp-based protocols because they will be able to incorporate the technology end to end in their systems in parallel with the efforts of standards bodies and working groups, although these groups will most certainly play an important role in the long term.

Begin at the Edge

Fundamentally, the need for chirp-based protocols and the networking architecture to support them starts at the end device sensors and actuators that cannot use IPv6 for connectivity to the Internet of Things for the reasons of cost and complexity *and* that require self-classification that would be unwieldy in IP in any case. As described in Chapter 6, classification of these chirp–based end devices by type and function will take place via an extensible marker system carried within the chirp packet and will be easily visible as these packets transit the network.

Initially, the use of chirp classification categories could be proprietary or vendor-specific for an OEM supplying both the end devices and the integrator function/big data services, but the classifications will rapidly be formalized *across* organizations. (See more details

of how these classifications could be created and managed in the following "Working in Groups" section). Once data streams are encoded in chirps with category classifications as to their type included, the data is inherently publish-ready, and some of the scaling benefits of the emerging IoT architecture can be seen.

CHIRPS IN IP PACKETS? WHY?

In advance of the proliferation of native chirp-based networks, the chirp information could also be specified within the payload section of a traditional IP packet by an "adapter" propagator node, which would encapsulate simpler terse data in the form of chirps with their inherent classification. This would allow subscribing big data systems to incorporate this information immediately and make possible a migration to the integrator function systems described in Chapter 5. The legacy IP packet containing chirp-formatted data will still need routing to reach a point-to-point destination, where the software is capable of deciphering the payload and acting on the data. That will likely be the first place where chirp protocols will be deployed.

The outgoing IP stream from the adapter propagator node could be Wi-Fi standards–based (i.e., 802.11). On the incoming chirp streams, the transceivers and their device drivers would need to look like ports on a local area network (LAN) switch for the Layer 2 hierarchical switch stack analogy to hold water. As long as the chirp device drivers on the adapter propagator node look and feel like IP "ports" on a legacy 802.11 access point (AP) "switch," multiple types of streams can be supported within the same AP. Alternately, network appliances may be installed to provide the chirp-to-IP interface, using Wi-Fi as a means to connect to the legacy IP network.

This technique provides one means of integrating chirp streams into legacy big data systems and may be an important transition path in the early days of chirp end devices. But it does not provide many of the other benefits of true chirp-based protocols such as broader data neighborhoods free of predefined IP peer-to-peer relationships and the tighter control loops made possible by distributing intelligence closer to the end devices in the form of publishing agents and localized integrator functions within propagator nodes. (Note that these limitations would be in place regardless of whether chirp protocols or IP are used.) The benefits of richer information usage and better control loops are much more attainable in native chirp networking and become even more compelling as the number of devices increases exponentially at the edge of the network.

In the long term, most propagator node/AP combinations will have support for native chirps and legacy IP built-in (see the following "Propagator Nodes Provide the 'And'" section), but other transitional APs could be imagined that provide powered USB sockets for device manufacturers to provide the chirp interface separately that are tuned for the specific chirp devices that they manufacture.

Making a Mark

In order to increase applicability of their end devices (and thus increase revenue), multiple suppliers of the same type of appliance, sensor, or actuator will be motivated to use the same formats in expressing their chirp data. It will thus be possible for their end devices to be incorporated across a broader range of integrator functions (from many suppliers) and in so doing, increase the number of potential applications.

Note that the chirp protocol uses both public and private sections, each with its own markers. Thus manufacturer-specific information and vendor-specific data can be safely represented within the same public category classifications. So although a marker of (for example) 6.8.11 might be used for a general category of moisture sensors, additional proprietary data within private segments of the chirps might specify vendor-specific features. In this form of incremental markers and meanings, a broad range of integrator functions provided by many different manufacturers and in support of different applications might add this moisture sensor chirp data stream to their information "neighborhood" and obtain some minimal data. This could take place even if the subscribing application was unknown to or even unthought-of by the organization originally deploying the moisture sensor.

But additional data might be included in a private section of the chirp, accessible only to integrator functions and other distributed intelligence in the network that possessed the correct "key." In our theoretical case, salinity or acidity might also be measured by the same sensor, but information on those parameters would be transported in proprietary private data segments within the same chirp packet as are the "generic" moisture readings.

Acting on Markers

Multiple intelligent agents may thus be acting on different strings within the chirp packet. The common propagator node operation may simply prune and bundle chirp streams into small data flows published to a wide variety of potential subscribers. Again, these subscribers may have the key to the proprietary additional data—or they may not.

In other specific propagator nodes, publishing agents may be biased by particular integrator functions to peer deeper into the private payload section and perform a more customized next level of routing and processing. This might include preferential routing to specific integrator function locations, "spoofing" by emulating round-trip acknowledgments locally, setting up specific forwarding bus timings or lower-level control loops, and so on.

Propagator Nodes Provide the "and"

In the early days, chirp-enabled devices will be the minority traffic on the Internet of Things. Simply because of the extensive installed base, large numbers of IP-equipped end devices will need to be accommodated as well. For that reason, many first-wave propagator node implementations will provide both chirp-ready *and* legacy IP connections such as Ethernet and Wi-Fi.

This emerging new class of hybrid devices will use chirp- and IP protocols interchangeably. These ambidextrous network elements will appear as two logically distinct devices, even if they are using the same transceivers (e.g., 2.4GHz unlicensed band radios). The added advantage of these IP-equipped devices is that they will also often have the processing power to house publishing agents, as required.

The input of these devices will be of three possible types as shown in Figure 8-2. Some IP packets will be the unmodified legacy IP streams from traditional devices. A second possible type (as noted in the sidebar "Chirps in IP Packets? Why?") will be encapsulated chirp streams within IP packets, intended for big data servers that are not yet fully chirp-aware. And a third class will be native emerging IoT architecture chirp data streams. This latter packet type will be intended specifically for chirp-aware integrator functions. Depending on the needs of the servers at the final destination, the transition propagator node will aggregate small data flows of chirp streams into IP packets or will simply pass them through legacy IP packets.

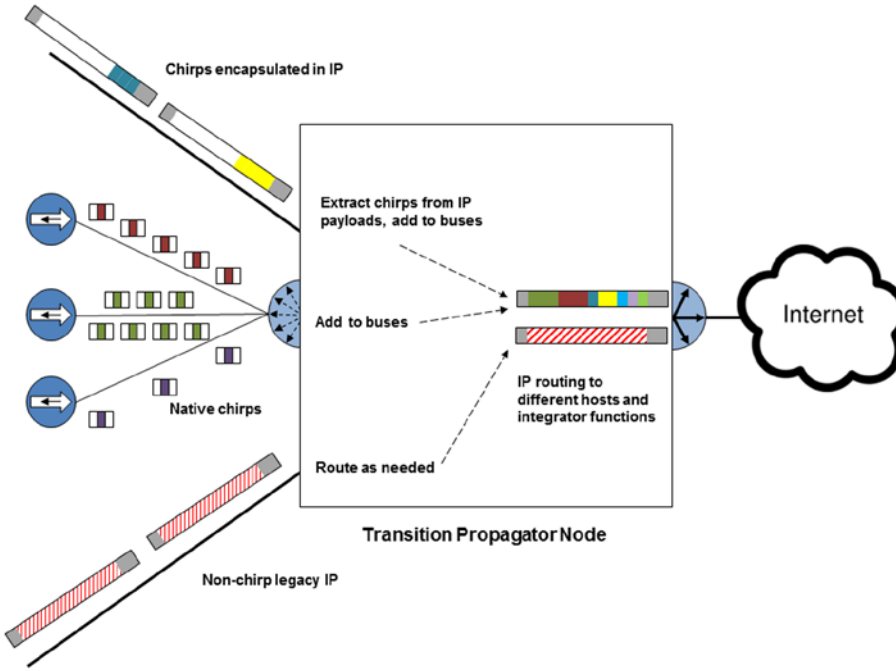

Figure 8-2. *Hybrid transition propagator nodes will handle legacy IP traffic, encapsulated chirp traffic, and native IoT chirps aggregated into small data streams*

As noted elsewhere, there will be many different packaging options for propagator nodes, including some with integrator functions on board that might handle some analysis and control tasks for their associated chirp end devices.

Because of their key role in translating and merging both legacy and emerging networks, transition propagator nodes of this type will necessarily be one of the first examples of equipment to be developed and marketed along with the first chirp-enabled end devices. Although some initial applications may be proprietary and OEM-vendor-specific, it is expected that more generic versions will also appear rapidly.

Open-Source Networking Solutions

One key to accelerating the development and proliferation of these translating generic propagator nodes will be taking full advantage of open-source technologies. One likely base (among a number of possibilities) upon which to build propagator node functions is OpenWrt, an operating system/embedded operating system based on the Linux kernel and primarily used on embedded devices to route network traffic. A chirp-enabled branch of this code could be produced quickly to allow rapid development of new propagator nodes, along with immediate integration into existing networking equipment operating under OpenWrt.

Gaining Access

Wi-Fi access points are one of the most numerous deployed networking solutions today, allowing a variety of devices equipped with 802.11 wireless capability to be connected into a network (today, nearly always IP-based). As such, they represent an attractive candidate for replacement by transition propagator nodes from a network topology standpoint. Virtually none of today's deployed APs supports the type of secure application layer and field upgradeability needed to incorporate chirp–enabled propagator node software directly.

But a new combined AP/propagator node device (likely based on OpenWrt) will include both traditional AP and IoT chirp–enabled propagator node capabilities, as seen in Figure 8-3. One key will be making the propagator node portion of the combined device "responsible" for both legacy and chirp communications to ensure that no changes are required for legacy IP IoT devices or big data servers. Multiple forms of connectivity will be made available over many different interfaces (e.g., Wi-Fi, IR, Bluetooth, Power Line, etc.).

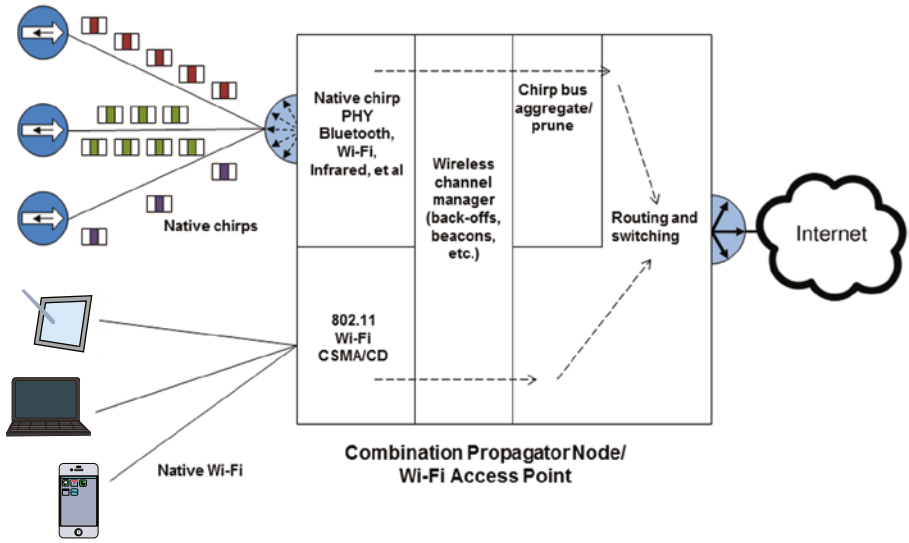

Figure 8-3. *Combination propagator node/AP devices will be an efficient means of merging traditional IP data with IoT chirp streams, sharing a single connection to the global Internet*

Clusters of simple chirp devices, currently not even imagined, will "connect" via these interfaces, with propagator nodes tasked to do the heavy lifting needed for conversion to small data streams, including routing and delivery via the logical "bus" described in Chapter 6. Much of this will occur without the need for arduous standards body consensus—at least initially (see the following "The Standards Conundrum" section). The chirp-enabled propagator nodes will integrate smoothly with existing IP devices and use the existing global Internet for transport. Even if chirp end devices use the same wireless frequencies as IP traffic (e.g., unlicensed bands), the propagator nodes will take over the timing and beaconing of all the wireless interfaces (both chirp-and IP-based), enforcing time slot reservations ensuring that chirp- and IP devices don't "speak" at the same time using existing capabilities within 802.11. Collaborative coexistence will be supported at all times within the emerging ecosystem because the propagator node/AP units are both chirp- and IP-aware.

It is hoped that using an open-source software model (see the previous discussion) for the development of propagator node capabilities may make it relatively straightforward for at least some existing AP manufacturers to quickly provide combined propagator node/AP units. These manufacturers would have the capability to extend the AP functionality to include an applications layer and also provide the device-layer abstractions so that new chirp devices can be supported with a "standard" interface to the chirp-to-IP bridge.

The Standards Conundrum

In the longer term, it is expected that a variety of standards bodies and working groups will formalize the specifics of the chirp packet and other elements of the emerging Internet of Things architecture. But the impending explosion in the growth of the IoT means that there is no time to wait for a drawn-out standard process before beginning to deploy this architecture. So a two-pronged approach will be necessary: de facto standards, working groups, and recommended practices allowing products to be brought to market quickly; along with a longer-term standards effort to codify these practices into standards. An example may be drawn from earlier machine-to-machine technology developments.

Machine-to-machine (M2M) communications are not new. Factory automation (e.g., robots, "intelligent" machines) has thrived on tight sensor–actuator control loops, where myriad sensors "feed" into Programmable Logic Controllers (PLCs) through the wired analog and digital I/O ports of the PLC controller. Relatively simple rule-based logic has been used to control complex machines composed of hundreds of sensors and actuators. The "circuits" turn on, based on logical switches turning on or off based on sensor data. When a circuit turns "on," actuators are activated. As a simple example, turning on a light switch closes a circuit to send electricity to a light bulb. Multiple such circuits, running concurrently within the PLC, have and do coordinate complex manufacturing processes.

These M2M communications and the tight control loops resulting from the custom-programmed circuits have clearly demonstrated the ability to generate complex competence from simple end devices such as sensors and actuators. Protocols and device drivers are often created by application software developers to meet the requirements of the specific process control required. A thriving manufacturing industry has evolved over the last two decades, based on proprietary, purpose-built, and terse sensor-actuator communications.

Standards existed for these sensors and actuators, but they were often home-grown by the sensor and actuator manufacturers, many times through Special Interest Groups (SIGs) within larger communities such as IEEE. However, because the device communications were local and entirely within a small community (e.g., a manufacturing line), there was no need for an overarching standard such as IP. In addition, in most cases the sensors/actuators are directly wired to the PLC controllers. There is no shared wireless spectrum to negotiate.

As more M2M sensors and actuators become wireless, sharing the same "air space" (i.e., unlicensed radio frequency spectrum) will become a challenge. Standard protocols such as ZigBee and Bluetooth evolved to support smaller communities of devices. However, all such devices were intended largely for human consumption of information and therefore were IP-based. They are currently being used to connect devices as part of a home audio system or home lighting system, being controlled by a home user's computer or smartphone. Note that they are human-in-the-loop systems; they are intended for humans to more conveniently control their environment, using the smartphone, for example, to remotely connect to their home lighting/heating systems or to link external keyboards or headphones to computers.

Machine-to-Machine Communications and Autonomy

More autonomous systems have evolved, where needed, to support more complex interactions from machine to machine and the machine with its environment. Although the human is still in the loop in a high-level control or advisory capacity, the devices are required to take more control in order to free up the human to do other tasks or because the human cannot respond adequately or in time (see Figure 7-3). This is exacerbated by the round-trip delays introduced in typical IP point-to-point relationships. By decoupling control loops, the emerging Internet of Things allows for rapid autonomous action near the edges of the network while still allowing long-term trends to be analyzed and overall control to take place at a higher level.

As described in earlier chapters, existing legacy protocols were originally intended for host-to-host or human-to-host conversations, not for the terse (and predominantly one-way) exchanges between myriad simple chirping end devices and big data integrator functions. But chirps will become the prevalent form of M2M communications in the IoT. Just as birds don't need to learn a common language to communicate effectively across the same medium (the air), so the end devices in the IoT may use only simple chirps optimized for their classification and function, counting on propagator nodes to make the conversions needed to allow use of the global Internet as the communications backbone.

It is simpler to delegate to these propagator nodes the task of performing translations across end device communities than to force everyone to use the same overly complex (and over-featured) legacy protocol formats. Overarching standards become less relevant as information neighborhoods become smarter at what they do within their areas of expertise. Autonomy and local control loops will also be much easier to operate and maintain without the IP overhead and round-trip communication necessary in legacy networks. This is another argument for simple and specialized chirp-based conversations between machines.

Shared Vocabularies and de facto Standards

In the machine-to-machine manufacturing application examples, the systems that currently use simplified communications schemes are generally private. In the emerging Internet of Things, publishing *and subscribing* to data streams is the primary activity, so obviously there is a critical need for shared vocabularies. A simple *but open* scheme, such as chirp-based networking, provides the potential for tremendous economies of scale in place of private vocabularies.

Networking standards such as IP were based on communication protocols at the lower level of routing and networking without specifying payload vocabularies. As long as the IP packet headers were universally understood, the payload portion of the packet would be routed correctly to the requested destination. The contents of the payload were decipherable by the recipient at the destination address; everything else served primarily as indicators of a routing infrastructure.

Because many agents will be performing similar tasks, shared networking techniques and payload vocabularies within application segments (e.g., moisture sensors) will engender reusability of data. Thus major OEMS such as General Electric, Samsung, Siemens, and Honeywell (among many others) may cooperate on the chirp protocol for products that overlap in functionality as a first level of interoperability.

This cooperation may also extend to some common functionality between OEMs in the publishing agent resident on some classes of propagator nodes. Although it would require a great deal of coordinated collaboration, it also would reduce the overall complexity of the system. Because the publishers and subscribers for similar devices will share common interests, there is value in sharing the same computing resources resident on the propagator nodes.

Propagator nodes are operating close to the edge of the network, so using the same publishing agents makes things simpler. Through the common vocabulary of similar devices, a new form of standards will emerge: one that is more focused on communicating state information versus networking/routing flow. Hegemonies exist within application segments in which collaboration is implicit. For example, the same repair centers service multiple types of home appliances (e.g., washing machines) from competing brands, or multiple pieces of different equipment at one site (see Figure 8-4). Providing the same vocabulary for diagnostics would make it simpler for a repair staffer to do the work.

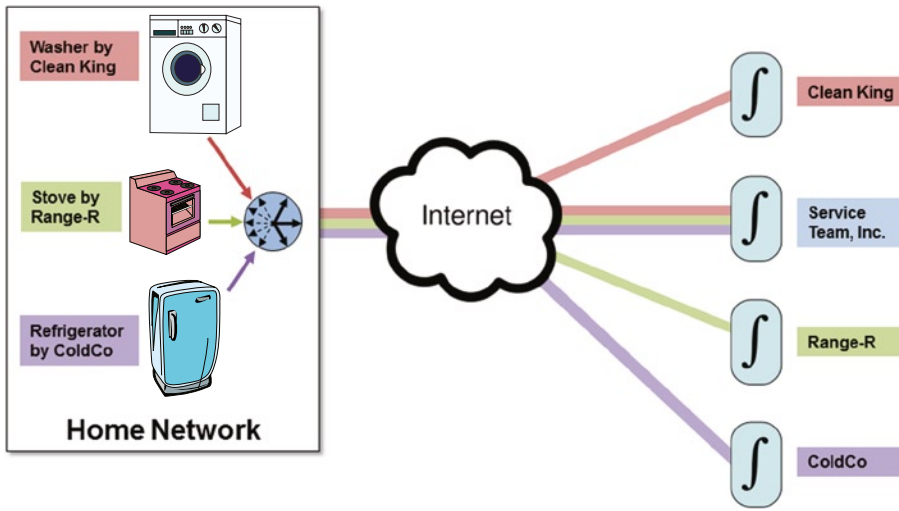

Figure 8-4. *In some cases, multiple networks may share information and network elements. Here, three types of machines from different manufacturers report usage and trends to independent integrator functions for each manufacturer, but they share status and alarm reports to a common third-party service company*

In time, sophisticated combined subsystems of analysis and control may develop organically near the edge of the network. These systems of systems, each capable of functioning autonomously, will increasingly continue to do so. Humans will be in the loop only for analysis of trends or periodic tuning and tweaking.

Build it and End Devices will Come

The explosion of smart devices (e.g., smartphones, home automation products) occurred because the support infrastructure was both prevalent and inexpensive. Internet connectivity became ubiquitous, at least in developed countries. This ready Internet access connected the lower–level consumer products to the higher end of cloud services and their applications.

A three-tiered ecosystem emerged: at the top, cloud-based applications could be downloaded to devices (computers and smartphones) via the middle layer of Internet connectivity, performed by an expanding network support infrastructure. Devices were thus "connected" to the cloud. New devices such as the Apple iPod were conceivable, in which the heavy lifting was performed by an intermediary computer connected to cloud applications. Some (agents, for example) could also run locally on the computer. In terms of the end device/propagator node/integrator function model of the IoT, end devices can similarly become widespread quickly when the network is there to support them.

In terms of a three-layered framework, at least two of three pieces must be available because only then would the cost of developing the third piece become economically viable. For example, iPods, with their limited inherent communications functionality (i.e., no IP stack), could not exist if computers running iTunes software did not exist as an intermediary or if the global Internet did not exist as a connection to cloud-based music services. In that framework, the "end device" (iPod) was supported by computer software downloaded to computers (propagator nodes within the IoT) connected to the cloud-based services via the Internet (the IoT's integrator functions).

OEM Leverage

In the legacy concept of the Internet of Things, IP is needed at each point (end device, networking element, and server). But for a chirp-based IoT to develop and proliferate, some use must be made of the existing elements to avoid the cost, complexity, and elapsed time necessary for a complete ground-up build-out.

OEM manufacturers are a likely first place where chirp-based disruptions would occur. OEMs are typically not interested in providing networking infrastructure, but their highest-end products (e.g., refrigerators, TVs) are becoming connected via IP. There is enough computing horsepower in these products to potentially serve as chirp–based propagator nodes for the OEM's large number of simpler, more lightweight devices that will never justify IP. The higher-cost devices would therefore support their less-sophisticated, chirp–based "country cousins."

There is incentive, therefore, to purchase a GE toaster if one owns a GE refrigerator, without burdening the toaster with its own IP connectivity. Or the presence of a Samsung TV would ensure that other Samsung devices, using low cost infrared transceivers (as in the TV remote), would coexist as part of the home entertainment system without each component requiring its own IP connectivity.

The "two-out-of-three" model makes sense for both manufacturers and consumers, as shown in Figure 8-5. Consumers pay less for their low-end devices (toasters) and their connectivity. Manufacturers can leverage their brands to provide interoperable families of products, all of which are connected in some fashion. In later years, they might potentially be updated via downloadable software to service chirp-based devices. And if desired, OEM manufacturers could use private markers and payloads in the chirp streams to lock-in buyers—although there will also be incentives to make public some or all of the information.

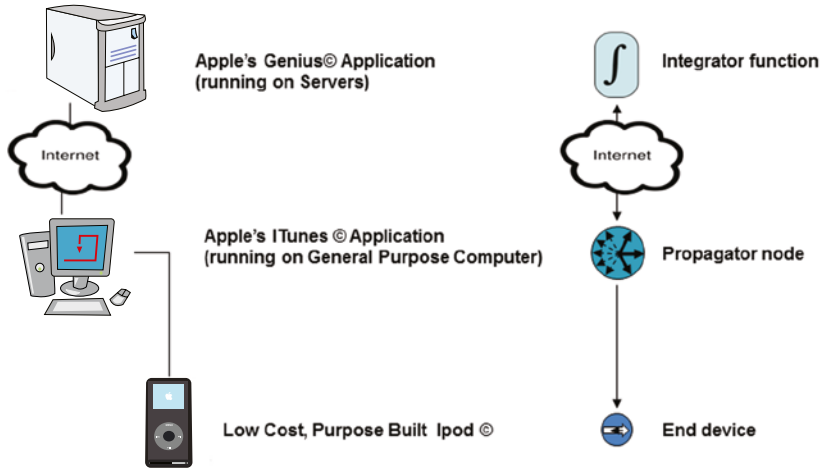

Figure 8-5. *Like a downloadable media player that is only economically viable if cost-effective computing power and global connectivity are already present, so an OEM's installed IP device and the global Internet might enable new low–cost end devices*

Applications-developer communities similar to Apple and Android application marketplaces will be encouraged to provide new applications for these newly connected devices. Ecosystems will emerge in which smarter IP-based products support their simpler chirp-enabled products. Giving away a free chirp-enabled toaster with every refrigerator purchase begins to make sense—the toaster becomes a useful device for control by the 'fridge. In this case, the refrigerator is the computer running applications on behalf of the toaster, which is still a purpose-built device. This mirrors the case of iTunes on a computer managing the simpler iPod in a previous generation of the three-layered ecosystem.

Shared Software and Business Process Vocabularies

Linux and its variants have become established as a primary embedded–system operating environment, largely due to open software initiatives. Proof-of-concept propagator nodes and publishing agents now being developed are currently based on Linux variants, and many future implementations will likely follow suit.

In the enterprise business world, Java is widely used for programming applications that may be written once and (theoretically, at least) used in many places. Programming in Java is simpler and more enterprise-business-process friendly. Translation mechanisms will evolve to convert business processes originally expressed in visual programming languages or in Java to simple rules that will be downloaded to integrator functions and/or the publishing agents on the propagator nodes. And this will be true for other enterprise software, as well.

Software as a Service (SaaS) has become a staple in cloud-based computing, and its counterpart in the Internet of Things may be a set of functions to be loaded on propagator nodes. Multiple propagator nodes from diverse manufacturers will need to connect and support a variety of big data services, so it is likely that the means to do so, including the translation mechanisms, will be made available as open source. Large enterprises and OEMs may use customized versions with proprietary protocols to access the private section of chirp protocols, but the ecosystem will support common vocabularies and processes to a large degree. Hence the semantics of an operation will be understood by the same category of devices, regardless of their brand.

The need to communicate in the same manner to big data cloud servers will drive common APIs and high–level control languages, as in the case of shared vocabularies. Although standards may emerge for these vocabularies in the long term, OEMS, working groups, and special interest groups will continue to promote this collaboration, driven by mutual interests and common practices.

Working in Groups

All in all, an organic process is expected for the development and deployment of the emerging Internet of Things architecture. But certain basic structures and tenants are keys to the success of the IoT. It is especially critical that the basic chirp structure be agreed upon and top-level classifications defined by a critical mass of IoT constituent organizations. The goal would be to reach a consensus rapidly on crucial parameters, permitting many companies and organizations to move quickly to develop their own products.

There are multiple alternate paths this development might take. One successful model is that followed by Bluetooth technology, which essentially began as a development within one company, but was shepherded by a handful of large companies collaborating as a special interest group. The time for successful interoperability testing and adoption of the technology was measured in years, however. The author favors a potentially more rapid approach, based on the open-source model (as seen with the OpenWrt distribution for Linux-based networking).

Whatever direction the initial development takes, the primary task will be definition of the highest levels of the chirp marker classification structure. It is anticipated that a one-byte first-order classification will provide a sufficient starting point for later added granularity. With these roughly *255* end-device categories set, working groups oriented toward specific industries could further define lower levels of addressing granularity. (Recall that the chirp marker structure is extensible to a very large numbers of classifications encompassing future needs.)

After the basics of the IoT are described, and products based on early versions of the definitions and parameters are being offered, it is likely that some standards body, such as the IEEE, will adopt chirp technologies into an existing standard as a working group or initiate a new standards effort. This would likely be driven by a larger player or OEM wishing to embrace standardization.

The rapidly expanding number of Internet of Things devices will create the need for this emerging technology in short order, so approaches that require minimum time to fruition are desirable.

Call to Constituencies for the IoT

Many different kinds of organization will have a stake in the success of the emerging Internet of Things. This section briefly describes what steps will be required of each of these constituencies.

Semiconductor Providers

Integrated Circuit (IC) "chirp chips" will be necessary for reasons of cost and power consumption at the end devices. Because of the minimal hardware and memory demands of the chirp protocol, the initial versions of these ICs may also be relatively simple, with greater integration, lower cost, and lower power consumption coming over time.

For propagator nodes, many off-the-shelf System-on-a-Chip (SoC) and System-in-a-Package solutions designed for data processing and network interfaces for traditional networking devices may be useful as building blocks, along with additional specialized ICs for the chirp "side" of the devices. For smaller packages in which publishing agents or integrator functions are incorporated, emerging compact devices such as Intel's Quark SoC may be preferred. Integrator functions will usually operate on general-purpose processors, and filter gateways may use existing router hardware.

The key challenge for semiconductor providers will be a quick determination of the specific parameters of the chirp protocol to allow rapid development. It is hoped that one or more semiconductor vendors will participate in early working groups and special interest groups.

Appliance and Other End Device Manufacturers

A number of sensor, actuator, and appliance manufacturers have already incorporated IP protocol stacks into their more sophisticated Internet of Things products. For these products, incorporating chirp self-classified data formats within IP payloads as an interim step toward the emerging IoT architecture may be a matter of software revision only. But the vast majority of end device types that will eventually be connected to the IoT do not yet have *any* network interface.

For these devices, the problem is somewhat chicken-and-egg: they will likely not be able to cost-effectively move forward until IC chirp chips are available for specific applications; and the semiconductor manufacturers may not move ahead rapidly with optimized chirp chips until the end devices are being developed. As noted in

"Major End-to-End OEMs" below, OEMs with a vested interest in end-to-end systems may develop the first wave of end devices with native chirp protocols, which may serve to accelerate broader deployment.

On the plus side, because the chirp protocol requires no central registry of network addresses (as the MAC IDs needed for Ethernet, 802.11, Bluetooth, and others), end device manufacturers may move quickly and independently to adopt chirp technology. Working from published top-level device-type classifications and the overall chirp packet structure, they may easily build devices that will interoperate with propagator nodes and integrator functions built by others.

Networking Equipment Vendors

Because the technology requirements are very similar, many of today's leading networking equipment vendors may move directly into the propagator node business. The only challenge may be philosophical rather than technical: a willingness to give up the mantra of "IPv6 everywhere" for the Internet of Things. The benefit, of course, is access to the new market to connect hundreds of billions of new devices. "Greenfield" markets are often more profitable than ongoing commoditizing sectors, so this alone may provide ample justification for investment.

But even vendors who steadfastly remain in the IP-only camp will still find their products used in expansions of the global Internet infrastructure needed between propagator nodes and integrator functions. Upon reaching the Internet, packets are packets – and the rising tide of the IoT will lift many boats. Existing IPv6 router devices may also be a good basis for the IoT filter gateways needed in some applications. In many cases, only configuration and programming will be needed.

Home Automation/Entertainment Suppliers

A tremendous potential exists for expansion of home networking in the form of chirp-enabled networking. One focus may be the TV set-top box (or a smart TV) that already increasingly includes Internet access. One can imagine future devices that connect not only to existing home equipment via infrared interfaces and the Internet via cable or Wi-Fi but also link the rest of the devices in the home via Power Line, Wi-Fi, or other technologies. Alternately, combination home propagator node/APs with the appropriate chirp transceivers built in will support both Wi-Fi IP and chirp traffic.

A local integrator function within the propagator node could provide the "brains" for home entertainment, climate control, security, energy management, and so on. Because this device will have access to a much broader set of devices *as well as other data sources* such as weather reports and utility updates, it will optimize the operation of the home as not previously possible. Unlike expensive proprietary solutions offered to date, proliferation of compatible chirp-enabled products will reduce costs, allow expansion over time, and eliminate reliance on single-vendor offerings.

Coordination with nascent standards work in the home automation space, and some integration or translation of existing open- or quasi-open-source technologies such as C-Bus, Insteon, KNX, X10, and ZigBee, will likely be important to acceptance of chirp-based end devices in the home.

Carriers and Big Data Providers

At the most basic level, major carriers will need to do nothing to support traffic from the emerging Internet of Things. Beyond the IP–equipped propagator nodes, traffic will be identical to all other Internet traffic and can be carried via the same backbone infrastructure. But there will be tremendous opportunities for cloud–based integrator functions, whether simply in the form of "power-by-the-hour" servers or value-added analysis and control services. The classification-based chirp protocol allows for preferential routing of specific small data flows, if desired.

Similarly, today's big data providers may integrate small data flows emanating from aggregated chirp data streams relatively straightforwardly with today's equipment and architectures. Big data customer optimization and the opportunity for new enhanced services will come as more propagator nodes are deployed that include on-board publishing agents. As big data providers move to the integrator function model for data analysis and control, they will be able to "bias" the distributed publishing agents (refer to Chapter 5) to allow independent local control loops for autonomous functions, as well as to tune the type, amount, and frequency of data being forwarded.

Major End-to-End OEMs

As mentioned earlier, one of the ways that long standardization cycles may be avoided in the implementation of the chirp-enabled Internet of Things is through the actions of large global Original Equipment Manufacturers (OEMs). Many of these OEMS already deliver solutions that reach from the edges of the enterprise or home to large centralized organizations. In many applications, these OEMs already use IP-powered networks extended by the global Internet to reach far-flung end devices, although data structures within the IP payload may be proprietary. But the emerging chirp-enabled architecture for the Internet of Things will benefit these OEMs in two ways.

The first and perhaps most obvious is that the cost (for processing, memory, power, and management) will be much lower for chirp-enabled end devices than for IP-enabled end devices. This cost savings will allow many more types and classes of equipment to be brought into the network, in which they may be monitored or controlled by the OEM systems. This extends the reach and differentiates lower-end equipment from generic competitors.

The second benefit is especially unique to the self-classified chirp traffic characteristic of the IoT: the capability to seek out and recruit *non-proprietary* data streams into an information neighborhood to provide added value to the OEM customer. The story is told of a global OEM that delivered a large robotic precision assembly system to India and set the machine up precisely as had been done in other parts of the world. Performance was poor with many breakdowns.

Eventually, an on-site engineer recognized that the higher ambient temperature was causing a deterioration of the low-viscosity lubricant called for in the manufacturer's specs developed in cooler climates. When this lubricant was replaced with a version more suitable for the environment, the equipment operated reliably. In earlier times, this sort of observation required an on-site human to make the observation and analysis.

But in the new world of the Internet of Things, the OEM might be able to recruit chirp data streams from existing nearby sensors that would provide temperature,

humidity, or other parameters that would help diagnose a fault condition at a distant installation. Because of the self-classified chirp protocol, these sensors could be installed by anyone, not necessarily the OEM. Unplanned and previously unknown data sources may be exploited along with data from the OEM's own equipment for a better experience for the end customer (refer to Figure 7-4).

Global Scope, Vast Numbers, Constant Adaptation, New Insights

As many have suggested in the past, it would certainly be *theoretically* possible for the Internet of Things to remain on traditional protocols such as IPv6. But for all the reasons described in this book, that path would close off the unprecedented potential of the Internet of Things. The scope is simply too large and the costs too great to expect traditional protocols to meet the need. Delaying deployment of a new architecture is no solution because it will never be possible to catch up.

The emerging Internet of Things architecture is designed to manage the unprecedented coming tsunami of data flowing to-and-from billions of end devices for applications both mundane and innovative. Lightweight self-identified protocols at the edge of the network, distributed networking intelligence, and ever-learning analysis and control functions will deliver on the promise of the IoT. Far from merely *addressing* billions of end points, this new architecture enables them to provide the information needed for powerful new knowledge, control, and efficiency in the final phase of the evolution of the Internet.