fer, although this is somewhat misleading. The primary motivation behind the OPP is that of exchanging electronic business cards. OPP uses OBEX, as do all of the object exchange profiles. The vCard object format mentioned in Chapter 9 can be used to represent a business card that can be exchanged using the OBEX protocol. Certainly other types of objects besides vCards (including notes, messages, calendar entries and in fact any object that could be exchanged using OBEX) can be used with the OPP, but the rationale behind the OPP is the business card exchange usage model.

The OPP assumes compliance with the GOEP and then further details the scenarios, functions and application considerations associated with object push. While it makes allowances for pushing (and in certain circumstances pulling) generic objects between a client and a server, the OPP focuses on pushing business card objects.

OPP Development

The OPP was the last of the object exchange profiles to be defined. Synchronization and file transfer are fundamental usage models that have existed since the SIG was formed; thus they were obvious profile candidates. Originally only these two profiles were defined within the object exchange family. It was not until January 1999 that the SIG made a distinction between two types of object exchange: a simple object push model and a folder-based browse, push and pull model. The former supports a "business card exchange" usage scenario, and this concept of unidirectional[3] object transfer eventually grew to become the OPP, although it was not originally called object push. The folder-based file transfer model is embodied in the file transfer profile, discussed below.

One key difference between the OPP and the file transfer profile is that the OPP instantiates a usage case in which data objects might be offered in an unsolicited fashion. File transfer (and synchronization too) normally are motivated by a desire of at least one party to acquire new or updated information, and they often involve user intervention. Pushing objects is a different model: at least one party might offer data without being asked, and that data is simply pushed to a static location (think of an in-box) without any application knowledge of file or directory structure. In some situations,[4] a user might configure her device to offer her electronic business card to any other device that comes within prox-

3. The previous footnote applies here also. The OPP can generally be thought of as a unidirectional object push, although limited object pulling is also possible, as explained in following sections.

imity and has the ability to take the business card.[5] This aspect is unique to the OPP and was one of the main reasons that business card exchange, or object push in general, was developed as a separate profile.

OPP Examined

Inheriting from the GOEP, the OPP defines a client and a server role, but further refines these roles to those of a *push client* and a *push server.* Just as in the GOEP, the push server is the device that provides the object exchange service, while the client is the device that does the pushing (and perhaps pulling) of objects. Of course these operations can be viewed as symmetric, in that if one device is pulling an object the other device might be considered to be pushing that same object. As explained for the GOEP, however, the OBEX model does distinguish between a client and a server, and the OPP maintains this distinction. As in the GOEP, the client and server roles do not imply anything about the underlying baseband master and slave roles.

One of the first concepts introduced in the OPP is that of pulling business cards, which for a profile dealing with pushing objects might seem unusual. Indeed, the OPP talks about push clients that can both push and pull objects to and from push servers; this apparent dichotomy merits further exploration. The key is found in the unique aspect of the OPP, noted above, that involves offering (pushing) unsolicited data objects. From the viewpoint of the push client, objects are always being pushed to a push server. In fact, during the errata process of the version 1.0 specification, a clarification was added to the OPP (present in the version 1.0B specification) explicitly stating that the push operation involves the client pushing an object to the server. Yet the same push client can also pull certain objects, as described below, from the push server. This concept is rooted in the OBEX transaction model, which has fundamental elements of pushing and pulling objects as well as the notion of a client and a server. One way that objects might be exchanged only through pushing is for a client to push an object to a server, then have the devices exchange roles, then have the new client (old server) push an object to the new server (old client). This would maintain a "push-only" purity but seems unnecessarily complex, given

4. Consider, for example, meetings and trade shows where business cards are exchanged; or any event at which a person might register herself by providing information typically contained in a business card (name, address, telephone number and so on).
5. The OPP does warn against automating the object push operation, though, and suggests that user intervention should be required to initiate the object push and pull functions.

that the underlying protocol supports both push and pull operations. Thus OBEX and the OPP allow a "push centric" usage model that also includes an optimization for pulling certain objects from a push server, although push servers are not required to support this feature. This optimization removes the need for a client-server role switch while still permitting the idea of a push-based usage model.

The OPP defines three functions: *object push*, *business card pull* and *business card exchange*. Object push is, of course, the fundamental operation and the only mandatory function within the OPP. The other two functions are optimizations of the type noted above, whereby the underlying OBEX pull operation is used to extract a specific object, namely the owner's business card, from the push server. The pull operation is optional for push servers to support; note that the pull operation in the OPP is restricted to pulling only owner business cards and not other objects, while the push operation can push any object (the file transfer profile, discussed below, provides a more general bidirectional object exchange). The business card exchange function is really just a composition of a business card object push and a business card pull function; thus it too is optional. Figure 14.2 illustrates the typical operation of the OPP.

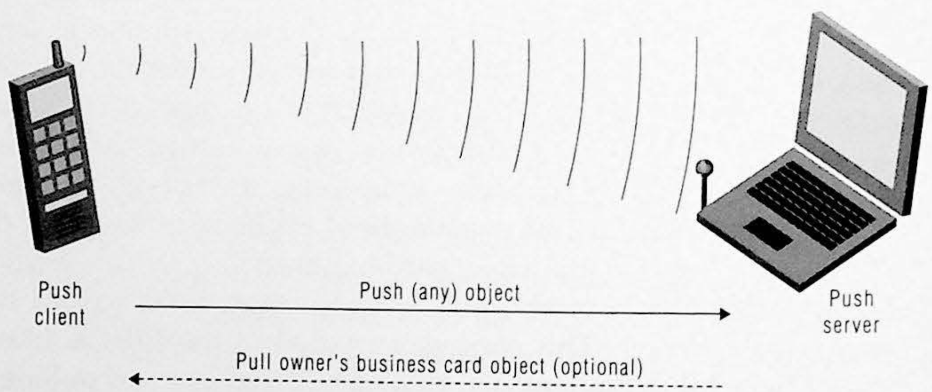

Figure 14.2
Object push profile typical operation. Note that business card exchange is just a business card push along with a business card pull.

Within the OPP are the procedures necessary to accomplish each of the object push, business card pull and business card exchange functions. The object push operation allows several types of objects to be pushed from the push client to the push server: vCard (version 2.1 of vCard is required), vCal, vMessage and vNote. The SDP service record of the OPP also allows a value for "any type of object," which would permit two devices to exchange objects other than those noted above, assuming that both devices understand the object format to be exchanged. The business card pull and business card exchange functions take advantage of the OBEX *default get object*, which is an object that can be pulled by type rather than by name. By specifying that the default get object contains the device owner's business card, the special case of pulling the default get object allows the business card pull and exchange operations to be accomplished within the context of the OPP.

The final point discussed here about the OPP is that of security. While exchanging objects between devices can be very useful, it also could be dangerous when one considers security exposures like viruses, violation of privacy and denial of service. All of these could be concerns if devices exchanged objects without any precautions. The OPP discusses at least two types of security precautions: the use of underlying Bluetooth transport security and user interaction. The OPP indicates that authentication and encryption at the baseband level must be supported, although they need not be used for every transaction. In addition, bonding (described in the GAP), which requires a trust relationship between the two involved devices, must be supported, but again need not be used for every transaction. If used, these security features can significantly reduce the exposures noted above, since objects can be exchanged securely, and only with devices that are known and trusted. Beyond this, the OPP mentions numerous times that user intervention is recommended for many of the steps required to accomplish object push and pull operations. The procedures in the OPP that outline each of the push, pull and exchange functions all include steps where it is recommended that the user decide whether or not to accept an object being pushed or to allow an object to be pulled.

OPP Usage

As discussed in the GOEP section above, middleware that implements the GOEP can provide a foundation for applications that implement the other object exchange profiles. Since all of these profiles share the common elements of the GOEP, it would not be uncommon to implement synchronization, file transfer and object push all in the same device. Much of the code–that which instantiates the GOEP–could likely be reused in each of the remaining profiles.

In fact, the OPP, from an application perspective, can be considered to be a special case of file transfer. Like file transfer (discussed below), the OPP pushes (and perhaps pulls) objects between a client and a server. The OPP restricts the types of objects that can be pushed and the circumstances under which specific objects can be pulled, so in some respects[6] it is a subset or restricted case of file transfer. Synchronization (also discussed below) could require significant additional logic beyond the object transfer functions, but file transfer and object push, in many cases, might be implemented in the same application (although the functions might be presented to the user as two separate applications).

Application considerations for OPP include the provision of a user interface to allow the required user intervention to occur. The user is the final arbiter of which objects are permitted to be pushed onto and pulled from his device, so the application needs to permit this sort of user interaction. Some of these functions might be candidates for integration with a general device control application such as the Bluetooth piconet minder application described in Chapter 8.

THE FILE TRANSFER PROFILE

The file transfer profile, or FP,[7] is the second of the three profiles in the object exchange family. Like synchronization and object push, the FP uses OBEX to exchange objects, in this case files and directories (or folders). During the early phases of the specification development, the definition of TCP/IP over Bluetooth links was investigated (see the discussion of OBEX over TCP/IP in Chapter 9) and thus the IETF file transfer protocol (FTP) was a candidate for a file transfer profile. In the end, the version 1.0 specification did not address generic IP networking over Bluetooth links, so FTP is not a part of the version 1.0 file transfer profile, although in the future this almost certainly will be an alternative method for file transfer. Within the version 1.0 realm, though, file and object transfer is via OBEX.

The FP can be considered to be a less restrictive, more robust form of the OPP in that it supports full bidirectional pushing and pulling of objects, yet it supports only two object types: *file* and *folder*. The FP does not directly address exchanging other object types like vCard, vCal and so on, although those object types could certainly be packaged as files and could be transferred using the FP.

6. At least from an implementer's view, although perhaps not from an end user's view.
7. We use FP rather than FTP to remove any confusion with the Internet file transfer protocol.

FP Development

The OBEX protocol was originally adopted from the IrDA to support the synchronization usage model. But OBEX also supports general file transfer, which has been used in IrDA for some time. File transfer has been a fundamental Bluetooth usage scenario since the SIG was formed, although it originally fell under the conference room scenario. As noted in the OPP discussion above, the FP originally was an all-encompassing profile for object exchange but eventually was split into the two distinct applications of object push, covered in the OPP, and folder-based browsing, pushing and pulling that remains in the FP.

FP Examined

Client and server roles are defined by the FP in a manner similar to the other object exchange profiles. In this case, the client is the device that initiates transactions and presumably will be pulling files from the server, although the client might also push objects to the server as described below. The server is the device that exports a folder to the client, which the client can browse to initiate requests to pull files (or other folders) from the server. The server also accepts other data from the client, including files that the client might push and requests to create or delete objects on the server. While the client and server role definitions are important for execution of the profile, many of the operations are symmetric, and it therefore seems likely that many devices can and will implement both client and server functions of the FP. Indeed, the FP notes that a device can support either role or both.

The operations defined by the FP are typical file manipulation operations, and they include:

- Pulling files and folders
- Pushing files and folders
- Browsing and navigating folders
- Deleting files and folders
- Creating new files and folders

Each of these operations is described from the client's viewpoint. Server operations occur in response to the client operations, and include supplying requested files and folders and responding to requests to delete and create objects. The folder browsing and file transfer (pushing and pulling) operations are mandatory for both client and server, and these allow the simplest form of file transfer, consisting of pulling a folder (the description, not the entire contents of all files in the folder), selecting one or more files in that folder, and pulling those files. Other operations that provide more advanced functions are optional; these include the ability to pull entire folder contents (which could be accomplished with an iteration that pulls all files in a folder, using just the basic folder browsing and file pulling operations) and the ability to create and delete objects. The FP includes procedures to follow for both client and server to accomplish these operations, along with the corresponding OBEX operations of the GOEP used in those procedures. Figure 14.3 depicts typical FP operation; two different types of devices are shown to illustrate that this usage scenario is not restricted only to traditional computers. Any two devices with compatible file representations could use the FP for file transfer.[8]

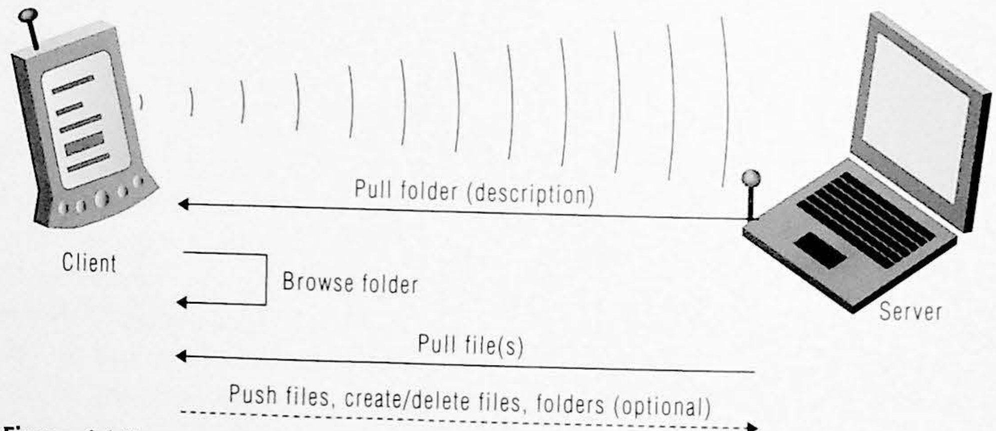

Figure 14.3
Typical file transfer profile operation.

The FP assumes user interaction for all file transfer operations. In addition to mandatory support (but optional use) of authentication and encryption, the FP mandates user intervention to initiate file transfers. As in the OPP, security exposures could surface when files are moved to a new device. Therefore the FP also requires user intervention to accept files from another device and to pull files from other devices. The FP assumes that a user interface will be presented on a device as a result of pulling a folder description. That user interface allows the user to browse and select files to pull; similarly, local files can be browsed and selected for pushing to another device. So while the protocol stack

8. Consider also devices such as digital cameras with diskette drives. Today pictures are transferred as files using the diskette. Since the camera must have a file system, the diskette drive could be removed and the files could be transferred using the Bluetooth FP.

includes security features that can be used in file transfer operations, the FP leaves to the end user the ultimate choice of which files to accept.

FP Usage

As described in the OPP section, it seems likely that both OPP and FP might be implemented together in many devices. Once the GOEP support is in place, the additions needed to support OPP and FPP are similar, and indeed might use the same code.

As noted in Chapter 3, file transfer is one of the most fundamental and useful functions of data networking. Many device manufacturers believe that transferring files and other objects is one of the most important scenarios to support in wireless communication, since most users are likely to expect and make use of this function. Thus the FP plays an important role in helping to ensure that file transfer can be accomplished in an interoperable fashion using Bluetooth technology.

Most devices that are likely to support the FP already have a file system and some sort of user interface for that file system; in addition they probably already include some notion of transferring files. While the mechanism used may or may not include OBEX file transfer, implementations of OBEX that meet the requirements of the GOEP can probably be procured or developed in a straightforward manner. With GOEP-compliant OBEX support in place, and with a Bluetooth adaptation layer in the device's software stack that permits the use of Bluetooth links, it should be possible in most cases to link (or perhaps adapt) existing file system user interfaces for use with the Bluetooth FP.

An integrated user interface for FP (and also for OPP) might include a Bluetooth piconet minder application like that described in Chapter 8 that allows users to select devices and services in proximity. One option for a device that is selected in this way could be to obtain file folders from that device and initiate file transfer (or business card exchange in the case of OPP as discussed above). This seems to provide an easy, straightforward and intuitive method for extending the existing functions of a given device or platform to take advantage of Bluetooth wireless communication between two cooperating devices.

The Synchronization Profile

Synchronization is a popular data communications application and it has been one of the Bluetooth usage models since the SIG was formed.

The final member of the object exchange family of profiles is the synchronization profile, or SP. The SP also builds upon the GOEP and uses the IrMC protocol to synchronize objects.

Synchronization can be considered to be a special case of object transfer in which programmatic decisions about which objects to transfer in which direction are made by synchronization software logic. The actual synchronization process can range from very simple (unidirectional pushing or pulling of a group of objects without any special treatment of those objects) to very involved (selective exchange of objects or even partial objects using principles like differencing and conflict resolution). Bluetooth synchronization as defined in the SP tends to more closely resemble the former, although application logic can be added to the basic operations of the SP to achieve more sophisticated synchronization models. Data can be synchronized between any two[9] entities, including devices and networks.

SP Development

Even though synchronization is probably the most complex of the object exchange scenarios, the development of the SP preceded that of the FP and OPP. The group that developed the IrDA interoperability protocols and their corresponding object exchange profiles was known within the SIG as the synchronization group.

Since the SIG's beginnings, synchronization among many classes of devices (phones, PDAs, notebook computers and others) has been a key usage case. In mid-1998, shortly after the SIG's formation, the possibilities for automated synchronization (described more fully in Chapter 3) had already been identified and the use of OBEX to accomplish these scenarios had already been proposed. The incorporation of OBEX into the Bluetooth protocol stack was primarily intended to support synchronization but, as we have seen, it also permits business card exchange, file transfer and other object transfer usage cases. A fundamental requirement was to be able to synchronize at least calendar and address book entries, although we will see that other data types can be synchronized as well.

The synchronization task force within the SIG was unusual (although not unique) in that, in addition to the five promoter companies, other contributing adopter companies (namely PUMATECH™

9. In fact it is possible to synchronize data among more than two devices, but we focus here on synchronizing between a pair of devices, which most closely matches the Bluetooth communication model.

and Extended Systems™, both of whose primary business is in the area of data synchronization) participated in the specification's development.

Because synchronization from the outset was one of the main usage models, the SP was one of the first profiles to be developed. It was not completed any sooner than most other profiles, though, owing mostly to the fact that new enhancements to the profile (like provision for automated synchronization and the addition of new and updated object formats) were added after it reached an initial level of stability. One interesting aspect of the SP's development is that it, along with the other object exchange profiles, was the first to add a section on service discovery, with service record and SDP transaction information. In this respect it served as a model for the other profiles, all of which (except for the "generic" ones) contain such information in the version 1.0 specification.

SP Examined

The SP first defines device roles, which once again derive from the GOEP and consist of a client and a server role. The client is the device that pulls data from the server, synchronizes that data against its own local objects, and pushes the resulting synchronized data back to the server. The server must support an object exchange service based upon the GOEP. As with the other object exchange profiles, these roles have no bearing on the underlying baseband master and slave roles. The SP indicates that the server is usually a phone or a PDA, with the client usually being a PC. This is curious, since servers traditionally are considered to be the more robust and capable machines. For the SP, it is the client that must contain the synchronization logic that determines how to process the objects to achieve a synchronized version of them. Thus the SP describes a PC as a typical client, since a PC is more likely to have available storage and processing power to operate a synchronization engine. However, this need not be the case–any device could take on the role of client or server, as appropriate. But today's typical synchronization models usually synchronize a "small" server, such as a PDA or phone, against a "large" client like a PC or even a network synchronization service. This model is preserved in the typical SP operation, which is illustrated in Figure 14.4.

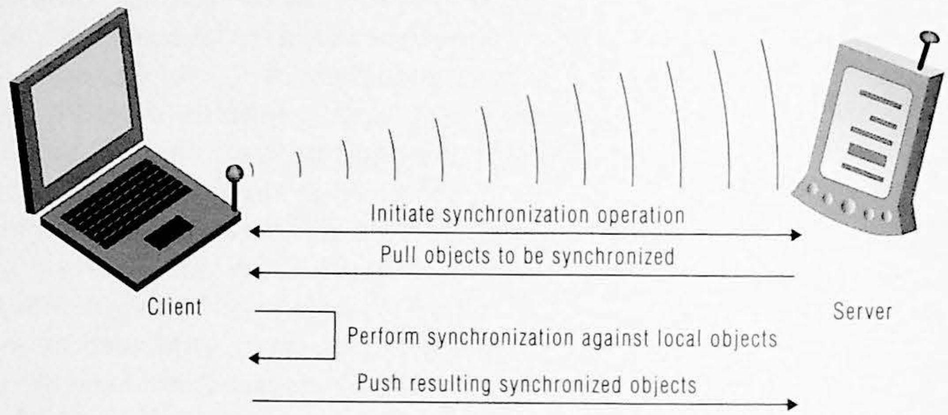

Figure 14.4
Typical synchronization profile operation.

The SP does not directly address the rules and processes necessary for a synchronization engine to actually synchronize the objects with each other. Instead it defers to the IrMC specification [IrDA99b] for that level of detail, since there is no particular reason to modify these aspects of synchronization for use in Bluetooth environments. Instead the SP focuses on the procedures needed to initiate and control the synchronization process. The SP discusses both client- and server-initiated synchronization and provides procedures to follow for these. In the former case there are two distinct scenarios: one for when the two devices are not yet known to each other and another for when the devices have already bonded (that is, are known to and trusted by each other). The second case is an optimization that takes advantage of the fact that the devices have already bonded. Another procedure is supplied for automated synchronization, which is a special case of client-initiated synchronization that is started without user intervention. Only bonded devices can synchronize automatically.

Several different object types can be synchronized using the SP; the profile does not mandate which object types must be supported. Instead it mandates that at least one of the defined object types–phonebook (or address book), calendar, notes and messages–be able to be synchronized. SDP is used to discover the supported object type(s) for the synchronization service.

The SP relies heavily on the GOEP and GAP, making use of several of the definitions and functions in each. In particular, some of the features defined in the GAP and GOEP are used for security. The SP mandates one of the highest security levels among all of the version 1.0 profiles. It restricts synchronization to bonded, or paired, devices and requires the use of authentication and encryption. Further, the OBEX-level authentication discussed in the GOEP optionally may also be used. Unlike the FP and OPP, the SP does not call for user intervention as a security measure. A user may initiate the synchronization transaction (although in the case of automated synchronization, even this user interaction is unnecessary) and may be informed of the status and results of the synchronization operation, and might even be consulted about desired actions (say, for conflict resolution) during synchronization. But the user typically does not authorize individual pushes and pulls of objects as in the FP and OPP (although the user might identify a set or class of objects that are to be synchronized). Since the SP does not rely on the user as a security arbiter, it specifies a relatively high level of other security functions from the protocol stack.

SP Usage

Synchronization is a somewhat specialized application, although a popular one. It is a bit different from file transfer and object push, although the SP uses many of the same underlying constructs and functions that the FP and OPP use. Since the SP builds upon the GOEP, an SP implementation could reuse OBEX middleware that is also used for FP and OPP. But unlike FP and OPP, which could be very similar (if not the same) applications, SP would not necessarily be expected to be implemented on all of the devices where FP and OPP are implemented. A digital camera, for example, might implement a file transfer function, as described in footnote 94 above, but the synchronization function on a camera seems less likely. As the SP notes, the devices most likely to use synchronization are notebook computers, phones and PDAs; these are devices that typically contain address books, appointments and other information (often called "PIM" or personal information management functions).

The SP does not provide any detailed description of the synchronization process itself; applications must look to the IrMC specification for guidance. Even beyond what is specified by IrMC, though, there is room for application differentiation. Synchronization applications can add value through enhanced user interfaces, optimized methods for exchanging and synchronizing objects and the like.

In the case of automated synchronization that is enabled by Bluetooth proximity networking, an application is still required, even though it may not be visible to the user at the time synchronization is performed. First, a user presumably will wish to configure his device for automated synchronization (just because this function is possible does not mean that every user will wish to take advantage of it; some users might want to use this feature selectively). In addition, some application software is needed on both the client and the server to discover the automated synchronization capability and to start and carry out the synchronization operation. In many situations it is also advisable to provide some sort of indication to the user that the automated synchronization is underway (this probably should at least be a user-configurable option, because many users are uncomfortable having their personal devices engage in nontrivial communications with other devices without their knowledge).

A final application consideration in the case of automated synchronization[10] is how to deal with a device that leaves the proximity range before the synchronization operation completes. Consider a case where a user walks into an office with a PDA in a pocket or purse. If appropriately configured, the PDA might begin synchronization with a PC in the office without the user being aware of it. The user might leave the office in the middle of the synchronization process. Thus the synchronization application needs to somehow account for this possibility, perhaps through a checkpoint and restart process of partial synchronization or some other means.

10. This consideration also applies for user-initiated synchronization, but less so. When a user initiates synchronization, she is probably likely to ensure that the devices being synchronized remain in range of each other until the operation completes. When the synchronization is started automatically, however, a user might not even be aware that it is occurring (see "Hidden Computing" in Chapter 3) and thus might very well walk into and then back out of proximity range before synchronization completes.

15

The Networking Profiles

The final group of version 1.0 profiles is what we term the networking profiles. This group consists of the LAN access, dial-up networking and fax profiles. As noted in the preceding chapters, each profile includes the aspect of a serial port, and the dial-up networking and fax profiles include an element of telephony. But to our way of thinking the primary focus of these profiles is on multihop (long-haul) data communications and networking. Clearly both dial-up networking and LAN access are intended to facilitate data networking, and the fax profile seems to have more in common with data networking (especially of the dial-up kind) than it does with voice telephony. All of the profiles in this group include an element of access to a wide area network for data communication.

All three of these profiles are intended to take advantage of Bluetooth wireless communication to make a well-known existing task easier by removing the need for cables. Using a fax or accessing a network, either directly or through a dial-up connection, are common tasks for many people. The profiles examined here define how to do these tasks in Bluetooth environments, without wires.

Each of these profiles, being more oriented to data than to voice, tends to be centered more around a computer of some sort than around a phone. However, just as the telephony profiles were applicable mostly to phones but also had aspects relevant to computers, the networking profiles are mostly for computers but also have aspects relevant for phones. Indeed, the dial-up networking and fax profiles can include both a computer and a mobile telephone, with the phone being used as a fax or data modem. So these profiles are expected to be most useful for, and most often implemented by, computers (stationary as well as mobile), but mobile telephones can provide services that gain them a key role in some instances of these profiles.

RELATIONSHIPS

As shown in the profile relationships diagram (refer back to Figure 11.1), all three of these profiles derive from the serial port profile, or SPP, that was described in the previous chapter. This is not surprising, since the SPP and its associated RFCOMM protocol are intended to allow legacy applications to make use of Bluetooth wireless transports, and all three of these profiles instantiate legacy applications (that is, they define how to do existing tasks without wires). So these profiles are a logical fit as members of the SPP family, which is the basis for the version 1.0 cable-replacement scenarios. They are also a good fit with the SPP technically, since all three profiles involve applications that most likely will include the notion of communicating over a serial port. In the case of dial-up networking and fax, the use of a serial interface is obvious, since both use a modem (or at least the abstraction of a modem) to communicate over a telephony network, and the most prevalent way to access nearly all modems is via a serial port. In the case of LAN access, the use of the serial interface might not be directly evident, since a direct network access cable is not necessarily modeled on a serial port. However, since the version 1.0 LAN access profile uses the point-to-point protocol (PPP), this sort of LAN access tends to resemble dial-up networking, and PPP maps well to a serial communication layer. Thus all three of the profiles derive from the SPP and use a serial port communication model.

The dial-up networking and LAN access profiles together make up the Internet bridge usage model. As described in Chapter 3, two similar yet different methods are defined for using Bluetooth links as a bridge to a larger network like the Internet. Those two methods are defined by the dial-up networking and LAN access profiles, respectively. Curiously, the fax profile has no specific publicized usage model behind it. So in a way the fax profile is not related to any of the other version 1.0 profiles except in being part of the SPP family tree. Indeed the fax profile is an example of the SIG's defining a formal specification for the use of Bluetooth links to perform easily understood usage models. In this respect the fax profile might be more similar to a printing or scanning profile, neither of which exists in the version 1.0 specification although they might be generated in the future. Since it does not derive from a common usage case, the fax profile is related only indirectly to any other version 1.0 profiles. It does, though, have some similarities with the other two networking profiles discussed in this chapter, which is why we include it here.

THE DIAL-UP NETWORKING PROFILE

The dial-up networking profile, or DUNP, specifically calls for both computing and telephony devices. Indeed, dial-up networking is an area where computing and communications overlap. In this case telephony devices access telephone networks so that computing devices can use that connection to access data networks. As compared to a typical wired scenario, the use of Bluetooth wireless communication could enable two kinds of cables to be replaced: one between the computer and the telephone and one between the telephone and the telephone line (assuming the use of a cellular mobile phone in the DUNP). Dial-up networking is possible with many mobile telephones today, without the use of Bluetooth technology, but normally a cable is needed between the computer and the telephone, even though the wide area network access via the mobile phone is wireless. The use of Bluetooth wireless communication removes the need for this last cable in dial-up networking, enabling a completely wireless[1] solution.

Dial-up networking involves the use of a telephone–in this case a mobile phone with Bluetooth technology–as a data modem. The computer uses the modem service of the telephone (probably unaware that a physical wired modem is not present) along with network dialing software to reach the network's access point that is connected to the (probably wired) telephone network. The DUNP even explicitly addresses the use of a physical modem, rather than a mobile telephone with a modem service, to perform dial-up networking. In this case such a modem would presumably be connected to a wired telephone line while providing a wireless Bluetooth link to its client(s). In this respect such a modem would resemble a voice or data access point (as are used in the cordless telephony and LAN access profiles, respectively) in that it offers a specific type of wireless access to some back-end network. Regardless of the physical device used, the computer is using a modem

1. No cables are needed for communications. We ignore wires that may be needed to supply electrical power, since most mobile devices can operate for some period of time untethered from an electrical power supply.

service to access a traditional telephone network that in turn offers access to a data network such as the Internet.

DUNP Development

The evolution of the DUNP (and its associated profile, the LAN access profile) is interesting. Originally there was a single Internet bridge profile, just as there is an Internet bridge usage model. As the marketing requirements document (MRD) was refined and additional thought was given to the topic of network access, two types of network access distinguished themselves. The MRD defined the concept of data access points and split these into two types: wide area network access, using modems, satellites, cellular networks and the like; and local area network access, using an access point to directly connect to an Ethernet, token-ring or similar LAN.

At first the Internet bridge profile attempted to cover both of these types of data network access. It later became clear that the two scenarios, while similar from an end-user perspective (and thus both considered to be part of the Internet bridge usage case), had different technical underpinnings and would probably require quite different implementations in devices. Thus the Internet bridge profile (like the three-in-one phone profile discussed in Chapter 13) was split into its two constituent parts: dial-up networking (now the DUNP) and LAN access (discussed below). The DUNP was developed by the telephony control task force within the SIG, since much of the profile dealt with telephone call control (recall that at this time a telephony control protocol called TCS-AT still existed–Chapter 10 discusses this topic further–and much of the DUNP deals with AT commands over RFCOMM to set up and manage the modem service). The LAN access profile, on the other hand, was developed by the SIG's networking task force, since telephony was not really relevant to that profile. Even though both of these profiles spring from a common usage model and they do have some technical similarities, they also have some differences at the implementation level.

DUNP Examined

The DUNP defines device roles for a *gateway* device and a *data terminal* device. The gateway is the device that offers a modem service to allow connection to the network that is being accessed; gateways typically are modems or mobile telephones. The data terminal is the device, usually a computer of some sort, that is using the modem service of the gateway device to access a data network. While dial-up networking is usually thought of from the viewpoint of the data terminal device accessing a network, the DUNP also notes that the reverse situation that allows the data terminal device to be dialed up via the modem device is also supported. Normally the data terminal device wishes to obtain access to a network, rather than to permit dial-up access to itself, although there are cases where the DUNP might be used for incoming connections. These gateway and data terminal device roles have no implications for the baseband master or slave roles.

Recall that while we consider the DUNP to be part of a networking group of profiles, it is a derivative of the serial port profile. The RFCOMM protocol is used to transport modem AT commands between the data terminal (computer) device and the gateway (phone or modem) device to establish and manage the connection to the network. The DUNP identifies a small subset of standard AT commands as defined in the ITU-T V.250 standard [ITU99] that are used to set up and manage the modem functions of the gateway device.

The DUNP addresses optional support for audio feedback. While its primary purpose is to support data calls, the audio capabilities of Bluetooth wireless communication permit a richer emulation of a wireline modem call by allowing for audio feedback. If audio feedback is supported, the modem tones associated with the call can be transmitted back to the data terminal device for playback through its audio output channel. Audio feedback can provide information to the end user about the call's progress, allowing the "squeaks and squawks" that many users have become accustomed to with dial-up networking also to be present, if desired, when using Bluetooth wireless links. The service record of the gateway's modem (dial-up networking) service indicates whether or not audio feedback is supported, so SDP can be used to determine whether or not this feature is available when the connection between the gateway and data terminal is established. Typical DUNP operation, including optional audio feedback, is depicted in Figure 15.1.

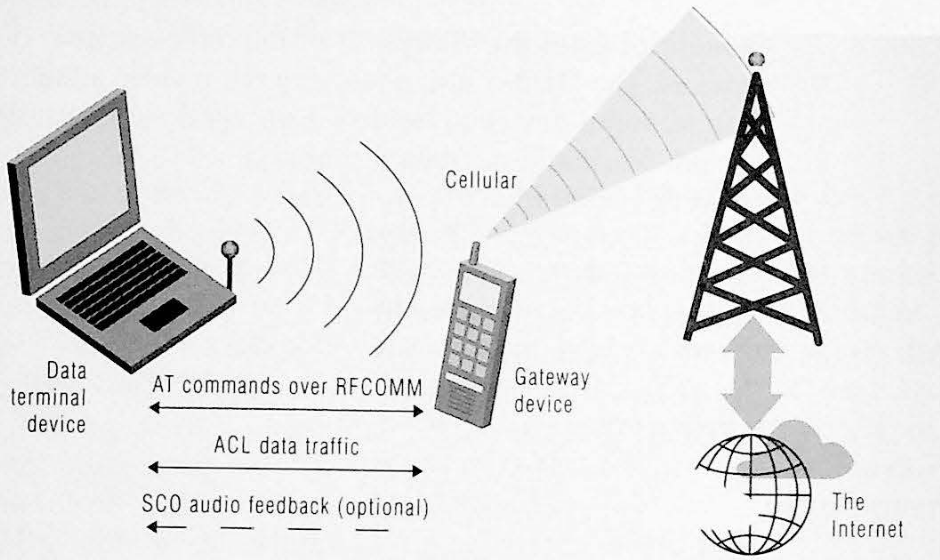

Figure 15.1
Typical dial-up networking profile operation. Note that the gateway device also could be a wireless modem access point connected to a wired telephone network.

Security in the DUNP is handled at several levels. Support for baseband pairing is mandatory, although it need not be used for every connection. However, since the most common case of dial-up networking with Bluetooth technology is likely to be a computer using a mobile phone to access the network, charges are likely to be incurred from the phone service provider for the wide-area cellular connection of the phone call. Thus pairing is advisable, since it can restrict use of the phone's modem service to known and trusted devices. A phone's owner may wish to make his phone's modem service available to his own computer(s) but probably not to anyone else's computer that happens to be in range. The DUNP also calls for the use of baseband authentication and encryption functions. At an application level, additional authentication such as a user identifier and password are likely to be required to access the network, once a connection to it is established.

DUNP Usage

The intent of the DUNP, like that of most other serial profile family members, is to enable legacy applications to take advantage of Bluetooth wireless links for existing functions. Dial-up networking is a common usage scenario, with many applications available for using modems to access networks. Through the use of RFCOMM serial port emulation and a minimal subset of well-known and standard AT commands, the DUNP provides dialer applications with a functional interface that is virtually identical to that which they use in the wired world. With the use of Bluetooth adaptation software, as described in Chapter 5, it should be possible for legacy applications to conform to the DUNP with little, if any, change.

Once dialing software has established the connection to the network, multitudes of applications that use standard networking protocols (like TCP/IP, HTTP, FTP and so on) can execute using the dial-up network link and the services available in the network. Hence the DUNP enables other applications such as browsers, e-mail and the like.

For more information about how the DUNP is used and how it relates to the LAN access profile, discussed below, see the following section "DUNP and LAP Compared."

The LAN Access Profile

The LAN access profile, or LAP, is the second profile that, along with the DUNP, instantiates the Internet bridge usage case. Like the DUNP, it uses established networking protocols over a Bluetooth wireless link to enable a computing device to obtain access to a data network. In the case of the LAP, a network *data access point* is used to connect to the network rather than a phone or modem used with a dial-up connection. Use of the LAP is analogous to directly connecting to a data network with an Ethernet (or similar) cable, although the usage is restricted to the use of the IETF *point-to-point protocol*, or *PPP*, over the Bluetooth link.

Like the DUNP, the LAP is based upon the SPP and is aimed at cable replacement. In this case the network access is local area, rather than wide area as in the DUNP. The LAP notes that this version 1.0 profile defines just one method for accessing networks via data access points, with others likely in the future (Chapter 16 describes some other Bluetooth LAN access possibilities). The most commonly described usage case for the LAP is for a computer to access a LAN, although the LAP does enable the development of data access points that could be used simultaneously by multiple Bluetooth clients. A specialized case of the LAP is its use to directly connect two devices for PPP communication between them rather than to access a larger network.

LAP Development

As noted in the DUNP discussion, the LAP was originally part of a single Internet bridge profile. When that profile was split, the LAP was developed by the SIG's networking task force. Even though the LAP and the DUNP both describe a method of realizing the Internet bridge usage case, they have some technical differences, because dial-up networking is a bit different from direct LAN access using PPP.[2]

The networking task force in the SIG was at one time called the TCP/IP task force, since its original mission was to define methods for traditional IP networking in Bluetooth environments. The group later decided that a more appropriate name for the task force was Bluetooth networking, since not all of the networking considerations were related to TCP/IP, although clearly IP networking is especially important. Unlike most other task forces, this group did not define any protocols in the stack but rather investigated ways to use traditional networking protocols, such as those defined by the IETF, over Bluetooth links. Thus there is no corresponding protocol stack layer in the core specification for the LAP. It is only this profile that defines LAN access with Bluetooth wireless communication, and it uses the RFCOMM protocol and the IETF's PPP. The LAP forms an important foundation for more robust future networking profiles, and it has always been considered to be an initial solution, but not an exclusive solution, for Bluetooth networking.

At first glance, it might appear that the solution developed by the networking group for LAN access is not as robust as might be expected, especially for peer-to-peer communication in LAN environments. However, the group's choice was not made lightly. This direction was chosen after long deliberations that considered the feasibility of other solutions, the time to market, and the time constraints for developing and publishing the version 1.0 specification. Peer-to-peer communication in purely ad hoc, wireless networks is an area that is not yet mature. While many industry and academic efforts are underway, there is no robust, fully tested, widely accepted method for achieving it, especially one that is suitable for resource-constrained devices like many of those used in Bluetooth wireless communication.

The SIG's networking group seriously contemplated IP networking issues like address assignment, name resolution, default router assignment, and so on. In an ad hoc network with no infrastructure services

2. Although, as the LAP notes, once a PPP connection is established between two devices, further interaction can occur much as it does for dial-up networking, which in turn often uses PPP internally.

such as DNS and DHCP, these and other necessary network support operations present unique problems. It became apparent that a solution to these problems could have a scope larger than just Bluetooth piconets. Had the SIG attempted to develop a general solution, it very likely would not have aligned with other industry activities that are proceeding toward maturity; furthermore, any such solution would have required prohibitive time investments in development and testing. Thus, a general solution that addresses the difficult issues associated with ad hoc IP networking (and that would enable several aspects of the conference table usage case described in Chapter 3) was deferred until after version 1.0. Instead, the networking group focused on developing the PPP-based LAN access solution–for two reasons: (1) PPP is a widely used Internet standard that addresses host configuration and preparation for IP communications; and (2) many devices, including popular PDAs, today support IP communications over PPP for dial-up access to IP networks. Hence, a large number of computing devices can support the Bluetooth LAN access profile without modification to their installed networking stack–a consideration that should not be overlooked.

Support for more general ad hoc peer-to-peer networking is an issue revisited by the SIG, as discussed in Chapter 16.

LAP Examined

The LAP defines device roles of a *LAN access point* and a *data terminal*. The LAN access point (which is just different terminology for a *data access point*; we use the latter term) is the device that exports PPP server function and is connected to a LAN, which might be Ethernet, token-ring or some other type of LAN.[3] The data access point is typically envisioned as a small "wireless plug"[4] connected to the wiring infrastructure of the LAN, and thus often mounted on a wall very much like a cabled data access point would be, perhaps in an office, a conference room, an auditorium or even in a home. The data terminal is the client of the data access point and thus contains PPP client function, which is used to establish the connection with the data access point that in turn permits access to the LAN. In the most general case, there is an association between these device roles and the baseband master and slave roles. Much as a cordless telephony gateway must be a piconet master

3. The LAP generally assumes the scenario of wireless access to a wired LAN, although access to a wireless LAN, such as one that uses 802.11 technology, is at least conceivable but might present some technical challenges.
4. Often informally called a *dongle*.

(as described in Chapters 10 and 13), a data access point must also assume the master role if it supports more than one data terminal client. In the case of only one data terminal client (for example, when the data access point is dedicated to a single client or when the LAP is used for PPP networking between two computers), it does not matter which device assumes the master role, but in general the data access point is assumed to be the master in LAP applications.

The use of PPP is key to the LAP. PPP is ideally suited for the connection between the data access point and the data terminal. IP network traffic (as well as other network protocols) can flow over the PPP link. PPP is also designed for use over serial connections, and thus within the LAP, PPP operates over RFCOMM.

The LAP is essentially a procedure for establishing a PPP connection between the data access point and the data terminal. Once the PPP connection is established, conventional IP solutions can be employed for networking functions such as obtaining an IP address. Standard IETF protocols can then flow over the PPP connection to access services on the LAN, much as in dial-up networking. Unlike dial-up networking, though, the PPP connection in the LAP is established directly over a packet-oriented data link and thus does not require modem functions and associated AT commands to establish the connection. Typical LAP operation with a single data terminal is shown in Figure 15.2; note that there could be multiple data terminals if supported by the data access point (in this case, each has its own separate Bluetooth transport and PPP connection to the data access point, and the data access point must be the piconet master). Figure 15.3 shows the special case of the LAP between two individual computers.

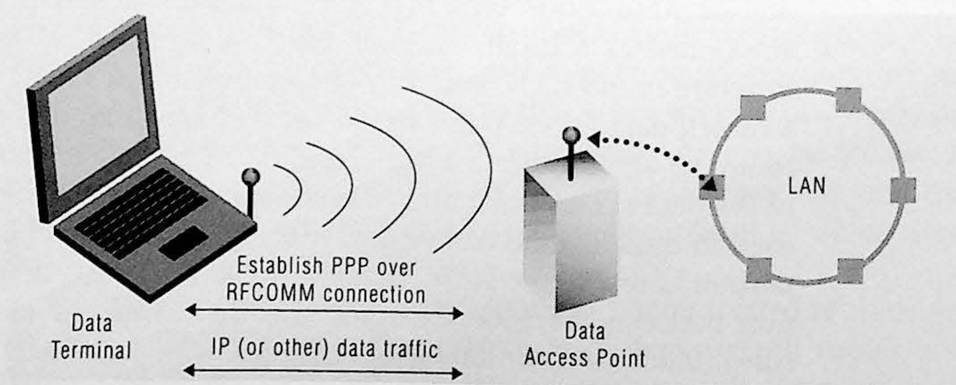

Figure 15.2
Typical LAN access profile operation. Note that data access points optionally may support multiple data terminal clients.

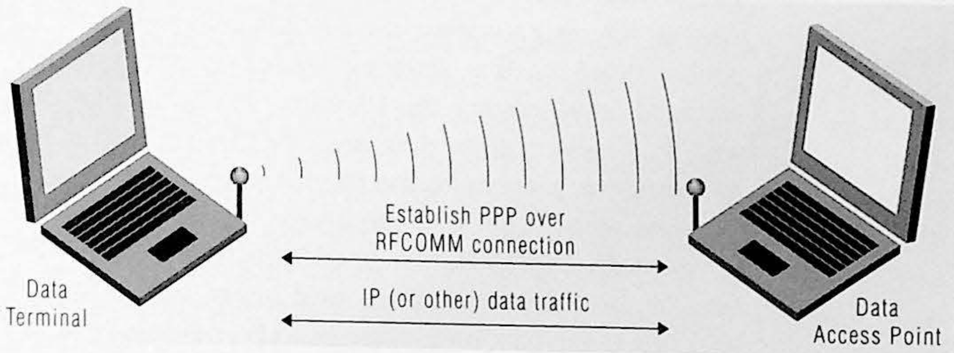

Figure 15.3
LAN access profile operation for networking between two devices. Either device can assume either role (data terminal or data access point).

The LAP has an interesting attribute that merits discussion. The service record associated with the PPP/RFCOMM service makes use of the *ServiceAvailability* attribute defined by SDP. The LAP is the only version 1.0 profile that specifies use of this attribute, although as a universal service attribute, it could be used by any service. A data access point could support many data terminal clients, and the ServiceAvailability attribute is used to indicate the degree of utilization of the data access point (how "busy" it is with current clients). Thus in the case where multiple data access points exist for a given LAN (perhaps in an auditorium or large conference room), a data terminal can perform SDP transactions with each data access point to locate one that has capacity to handle additional clients. The SDP specification defines the ServiceAvailability attribute generically without specifying particular values to indicate degrees of utilization. The LAP specifically defines values for ServiceAvailability for data access points,[5] with a percentage range that roughly corresponds to the number of possible active slaves in a piconet of which the data access point is master.

Security is a significant consideration of most networks; hence the LAP defines a high degree of security for the PPP connection that permits access to those networks. The LAP mandates authentication using device pairing, which can help to ensure that only authorized devices gain access to the network. This is important for corporate and other private networks; the network owner may not wish to grant network access to any device that happens to be within range of a data access

5. The actual values are found in the Bluetooth Assigned Numbers portion of the core specification (volume 1).

point (that device might belong to a visitor who is not authorized to access the private network). PIN-based authentication is required to authenticate with a data access point, so network access could be granted by divulging the PIN to the prospective data access point client. For more public data access points where it may be desirable to allow almost anyone to use the LAN, the PIN could be publicized to all persons who ought to have access, or–as the LAP specifies–a default, zero-length PIN could be used, effectively permitting universal admission to the data access point and hence to the LAN. In this latter case, additional security measures could be necessary to restrict access to services on the LAN or to other networks that may be accessible from the LAN. The LAP also insists upon the use of encryption of all the traffic on the Bluetooth wireless link between the data terminal(s) and the data access point. In addition, other higher-layer security mechanisms, including various PPP authentication schemes mentioned in the LAP, may be used to authenticate and authorize users of network resources.

LAP Usage

As with the DUNP, the motivation for the LAP is to access networks, primarily (but not exclusively) IP networks. So if middleware exists for PPP, along with a requisite IP stack, the same sorts of applications noted for the DUNP can execute over the LAP's PPP link: browsers, e-mail, FTP file transfers and so on. IP networking stacks and applications that use them are common in many devices, and these legacy applications are enabled for use with Bluetooth wireless communication via the PPP link defined by the LAP. Furthermore, other protocols can operate over the PPP link. Notable among these is WAP. The specification does not include a WAP profile; however, the use of a Bluetooth PPP/IP connection as a bearer for WAP traffic is described in volume 1 of the specification, and that part of the specification contains some information similar to that found in profiles. In particular, SDP service records are defined for WAP interoperability, and the PPP connection as defined by the LAP for IP traffic is exactly what is used for WAP network access.

In the next section we offer additional information on how the LAP is used as well as a comparison of the LAP usage versus that of the DUNP.

DUNP AND LAP COMPARED

Because the methods used to access IP-based services in the DUNP and LAP are similar, we assert that a data terminal device that implements both profiles could be developed with little more effort than would be required to implement just one. Moreover, the user experience for both of the profiles on such a device could be quite similar, with applications providing the same user interface and procedures for both the DUNP and the LAP. This is an added benefit to the user, who thus can be concerned with only the task at hand (perhaps browsing or accessing a corporate application), rather than with the underlying method used to connect to the data network.

For either of these two networking profiles, the ultimate objective is to enable a connection between the PPP client function in the data terminal device and a PPP server function residing at the edge of an IP network.[6] The primary difference in the two profiles is the role that the Bluetooth link plays in enabling this connection. Figures 15.4 and 15.5 highlight the differences and similarities in supporting IP communications using these two profiles, showing a typical protocol stack used in each. To connect, log in, and authenticate oneself to a PPP service, one may use a dialer application, like those used to connect to an Internet service provider over telephone networks. In the case of the DUNP, a modem connection is required to access the PPP server, and the Bluetooth link replaces a serial cable between the data terminal device and the gateway device that contains the modem service. In the case of the LAP no modems are involved, but the Bluetooth link is used as a substitute for a direct serial connection between the PPP client in the data terminal and the data access point that exports the PPP server function. Apart from this difference regarding the role of the Bluetooth link in the two profiles, the same applications and processes used to achieve IP connectivity with the DUNP can be reused in the LAP.

6. Note that other protocols besides IP can be multiplexed over PPP. Thus the IP discussion here applies for other protocols, too, but we focus on IP as the most commonly used networking protocol.

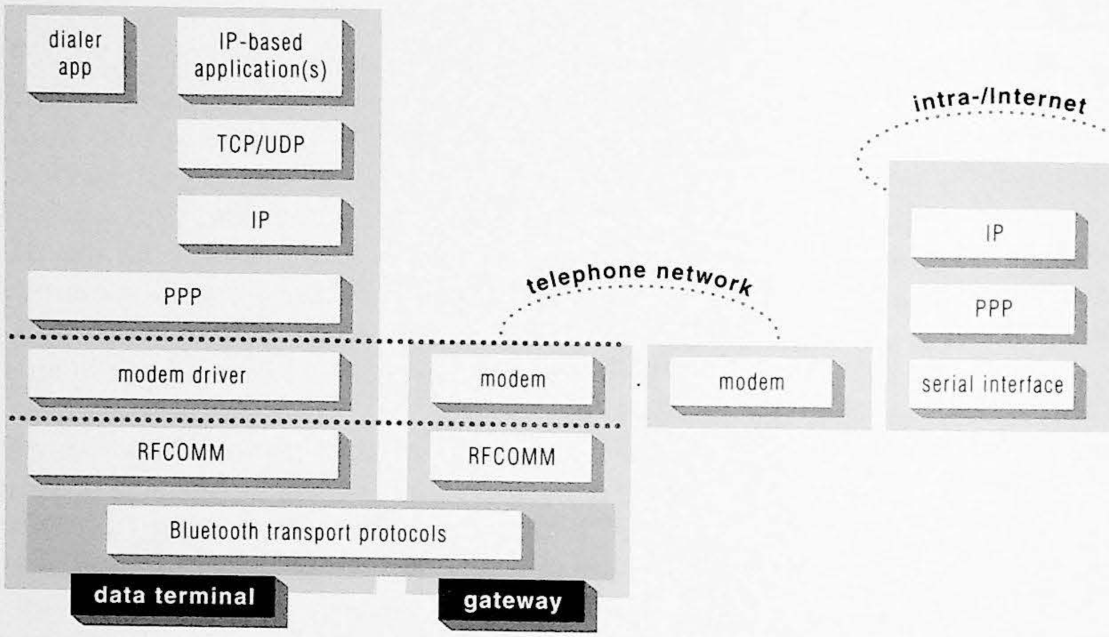

Figure 15.4
Use of a Bluetooth link in the dial-up networking profile (DUNP).

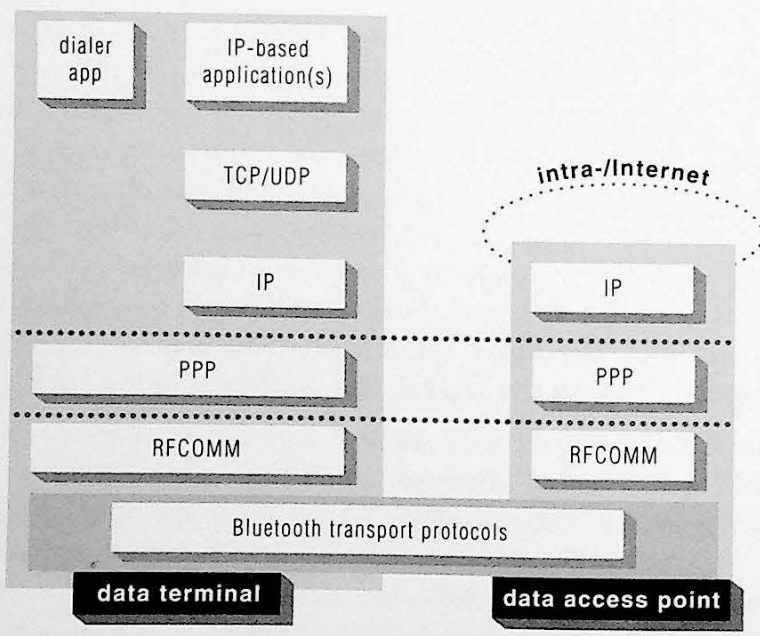

Figure 15.5
Use of a Bluetooth link in the LAN Access Profile (LAP).

Even though similar applications can be used with both profiles, one interesting usage scenario that differs between the LAP and the

DUNP is the aspect of mobility. In the typical case of the DUNP, the Bluetooth link is between the computer and the phone, while the network connection is carried over the phone's wide-area cellular connection. Since cellular networks use handoff technology to deal with changing locations of the phone, the network connection can be maintained even when the user is mobile, so long as the computer and the phone stay within proximity to each other (which is likely, since both are personal devices presumably kept with their user). With the LAP, though, the Bluetooth link is the network connection. So long as a user remains in proximity to the same data access point (as might occur in a conference room or auditorium), the network connection can be maintained. But if the user is mobile, the PPP connection to a given data access point could be terminated, requiring a new connection to be established to a new data access point in the new vicinity, even if both data access points are connected to the same LAN. The specification does not address any sort of handoff scheme for this scenario. Solutions do exist, but they must be implemented at the application layer of the client and/or in network middleware (which might or might not be directly associated with the data access point).

THE FAX PROFILE

The fax profile, or FaxP,[7] might be considered a special case of the DUNP (and in fact the DUNP mentions fax calls when it describes the data calls necessary for dial-up networking, but it states that fax is not part of the DUNP and instead is addressed in the FaxP). In many respects fax and data transmissions are similar: both modulate and demodulate commands and data between two endpoints over a telephone line. Yet there are differences, much as a data modem is distinguished from a fax modem, and there are special considerations for faxing over Bluetooth wireless links. Thus fax function is addressed in a separate profile, and even without a specific fax usage model, it is an assumed scenario due to its similarity to data calls.

FaxP Development

None of the published Bluetooth usage scenarios address fax function. The MRD[8] makes only two passing references to fax usage cases. So rather than having an associated usage model with a catchy name, the

7. We use the term FaxP to distinguish it from the file transfer profile that we call FP.

FaxP simply tells how to do wireless faxing using Bluetooth links. While we treat the FaxP here as a networking profile, its heritage is in the telephony-based profiles. As already noted, the DUNP and LAP were originally parts of a single Internet bridge profile. When the DUNP was made into its own separate profile, it initially included fax scenarios as well as dial-up data networking (this is evident from the many similarities in the structure and content of the two profiles, as well as the references to fax calls that remain in the DUNP). Soon thereafter, the FaxP was also split into its own profile based upon the considerations above that make fax its own distinctive usage case.

At about the same time that the FaxP was split into a separate profile, there was significant debate about the fax classes that could and should be supported over Bluetooth links. We do not present a detailed discussion of fax technology here,[9] but we do describe enough about fax classes to frame the issue regarding fax in Bluetooth wireless communication. In what is called *Group 3 fax*,[10] there are three protocols of interest within the FaxP context: class 1, class 2.0 and class 2. The former two are ITU-T standards while the latter is an industry de facto standard, and indeed class 2 and class 2.0 are different (although similar, and much of the following discussion of class 2.0 generally applies to class 2 also). The debate about fax class support in Bluetooth environments centers around timing requirements of these fax classes. A difference between class 1 and class 2.0 fax is the functional split of two major components of fax transmission: call control and image processing. In the typical FaxP usage case, a computer works with a mobile phone to send and receive fax information, similar to the typical DUNP case. In this typical configuration, both image processing and call control functions are performed by the computer for class 1 fax; whereas for class 2.0 the phone manages call control with the computer handling image processing. There was some concern within the SIG that the Bluetooth link between the computer and the phone could cause delays sufficient to violate some fax timing requirements. These concerns are most pronounced with class 1, where the computer must manage the call control functions over the Bluetooth link in addition to the image-processing load (the division of function between devices in class 2.0 makes it somewhat less susceptible to these timing violations). There

8. Recall from Chapter 4 that the marketing requirements document preceded the specification and defined the usage models that drove the requirements for protocols and profiles.
9. Interested readers can refer to, for example, [ITU96] as well as the standards listed in the "References" section of the Fax Profile chapter of the Bluetooth specification [BTSIG99], volume 2.
10. More properly, Group 3 facsimile, but we will continue to use the common term "fax."

was a proposal within the SIG to make support for class 1 fax optional based upon these concerns. After much study and debate, the SIG finally chose to mandate support for at least one of the three classes (1, 2 or 2.0), without specifying any particular one as mandatory to support, and without directly addressing the issue of timing requirements with regard to these classes (these considerations are left to the implementer, with the guidance of the Bluetooth specification and fax standards).

FaxP Examined

It is not surprising, given the history of profile development, that the FaxP is quite similar to the DUNP. The examination of the FaxP here is abbreviated, since much of the DUNP discussion also applies to the FaxP. The FaxP defines the same device roles as the DUNP, namely a gateway device (typically a mobile phone or a fax modem) and a data terminal device (typically a computer). These device roles have no bearing on the baseband master and slave device roles. Like in the DUNP, both outgoing calls (fax send) and incoming calls (fax receive) are permitted.

Since the FaxP is a derivative of the serial port profile, AT commands are used over an RFCOMM link for call control. The AT command set used is dependent upon the fax class(es) supported. Audio call progress feedback is optional and is handled in the same manner as with the DUNP. Typical FaxP operation is shown in Figure 15.6; note that the gateway device also could be a modem.

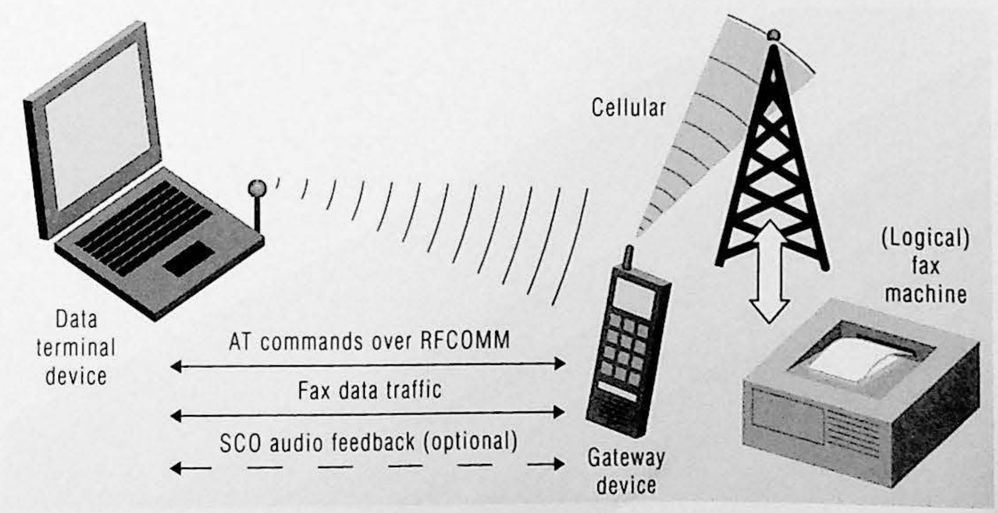

Figure 15.6
Typical fax profile operation.

The FaxP SDP service record is used to determine whether or not the optional audio feedback support is present, as well as to determine the fax class(es) supported, although the latter also can be determined using AT commands.

Because fax transmissions are generally considered reasonably secure, the FaxP mandates a relatively high level of security. Authentication and encryption are required, as is support for bonding and at least one of security modes 2 or 3 (described in Chapter 12).

FaxP Usage

Clearly the FaxP is another example of a profile to enable existing usage scenarios to be accomplished by existing applications in a wireless fashion. Fax technology is quite mature and is widely used today. Through the use of modem and serial port emulation, the FaxP is designed to allow legacy fax applications to operate over Bluetooth links with little, if any, change.

Part 4

THE FUTURE OF BLUETOOTH TECHNOLOGY

This book concludes with an examination of future directions for the Bluetooth technology. Chapter 16 discusses some possibilities for future applications of Bluetooth wireless communication, including topics addressed by the SIG subsequent to the publication of the version 1.0 specification. These include automotive, imaging, printing and other scenarios. We also discuss the product landscape and marketplace for Bluetooth devices in the year 2000 and beyond. Chapter 17 offers concluding remarks about present and future opportunities in this field.

Part 4 is intended to provide a snapshot of Bluetooth wireless communication in the year 2000 and a vision of where the technology may be headed in the future.

16

Beyond the Version 1.0 Specification

Having examined the version 1.0 specification in detail, we now turn our attention to post-version 1.0 matters. In particular we look into the activities of the SIG in developing new profiles, as well as some possibilities for products that use Bluetooth wireless technology. The thesis of this chapter is on new applications that might appear in the realm of Bluetooth technology, rather than on factual content that was explored in the main body of the book. Hence the tone of this chapter and the one that follows is a bit different.

Earlier in the book we noted that for version 1.0 the SIG focused on enabling basic cable-replacement scenarios. The SIG consciously decided to defer profile development that would support many more advanced yet interesting and valuable usage cases so as to accelerate the development of the version 1.0 specification. Many of these new usage models are addressed in profiles developed after version 1.0 publication. In this chapter we examine those profiles under development in the year 2000 along with other associated work within the SIG.

Just as the version 1.0 profiles are not a complete picture of what Bluetooth wireless technology offers, neither is the second round of profiles[1] to be published by the SIG the final answer about the specification. Indeed, the story for Bluetooth wireless communication may well be one without a definitive end, since industry innovation is likely to spawn new applications of the technology for years to come. In this chapter we examine just a few possibilities that go beyond the first two

1. Tentatively called version 2.0 of the specification, although this is subject to change before final publication.

editions of the specification. In conjunction with the SIG's specification work, we also explore the landscape of Bluetooth products, both those that are being marketed and those that are likely to appear in the foreseeable future.

THE SIG RECONSTITUTED

The SIG's original charter technically expired with the publication of the version 1.0 specification. That charter called for the specification's delivery, and once that was achieved, the SIG in one sense ceased to exist. The bylaws of the SIG allowed for the publication of errata to the original specification, and version 1.0B was published in December 1999 with many corrections and clarifications to the version 1.0A specification.

The SIG's work didn't really stop, though, after the initial specification was published. During the latter half of 1999, representatives of the original promoter members of the SIG held frequent discussions about the next steps that the SIG would take. These discussions culminated in December 1999 with the announcement of a newly chartered SIG which included four new promoter companies (3Com, Lucent, Microsoft and Motorola) in addition to the original five founding promoter companies (Ericsson, Intel, IBM, Nokia and Toshiba). Along with the new promoter group, the SIG also had a new organization, including a new class of members called *associates.* Associate members are somewhere between adopter and promoter members and may participate in specification development and SIG technical meetings. The associate membership category was created to permit broader participation in the SIG, and several companies immediately joined as associates. At the same time, the SIG also announced the formation of several new working groups, most of which are developing new profiles. This work, underway in 2000, is reviewed below.

New Working Groups and Profiles

Some important usage models were deferred during the development of the version 1.0 specification, and a number of new ideas for usage scenarios have surfaced as the Bluetooth technology has evolved. In 2000, the SIG chartered several new working groups to explore many such usage models, with most of them resulting in new profiles. A brief review of each working group underway in 2000 follows.

It should be noted that with the version 1.0 specification available and with many implementations proceeding based upon its contents, backward compatibility is a key concern of the SIG. All of the working groups include compatibility with the version 1.0 specification as one of their core objectives. Indeed, this is why most of the version 2.0 specification work is embodied as profiles: profiles provide a way to introduce new function without affecting those capabilities that already exist. All profiles, save the GAP, are optional. As new profiles become available, implementers may choose to support them without affecting existing functions. The protocols in the core specification are not expected to change significantly as the post-version 1.0 work proceeds; in some cases, optional extensions may be developed. But most new specification content will be delivered in the form of profiles.

Radio 2.0 and Coexistence Working Groups

Chaired by Ericsson and Nokia, the radio 2.0 working group investigates optional extensions to the radio specification. Among these are increased data rates, improvements to baseband functions (especially the inquiry process), "handoff" capability to support roaming, and better coexistence with other technologies operating in the 2.4 GHz spectrum.

Perhaps the most prominent feature under investigation by the radio 2.0 working group is that of higher data rates. The quest for more bandwidth in all types of communication seems insatiable, and most technologies are constantly striving for higher speeds and throughput. Increased data rates have been seen in both wired and wireless communication, with Ethernet and IrDA being but two examples. Bluetooth wireless technology is no exception, and many knowledgeable engineers believe that Bluetooth wireless communication can occur at higher speeds. In 2000, the radio 2.0 working group was looking into at least doubling the raw transmission speed of Bluetooth links to 2 Mbps, with some proposals that could increase data rates even more dramatically.

Like all of the working groups, the radio 2.0 group is concerned with backward compatibility. The radio 2.0 specification is expected to take the form of optional extensions. Fundamental principles of the Bluetooth radio, including global operation, low cost and short-range communication, will continue to be at the heart of the radio specification.

One particular radio consideration, that of harmony with other 2.4 GHz technologies, merits its own working group: the coexistence working group. This group is concerned with issues such as interference and performance impacts when multiple RF technologies are used in the same time and space. Working with other organizations, such as HomeRF™ and the IEEE 802.11 and 802.15 working groups, the coexistence working group produces recommendations to allow the various 2.4 GHz technologies to work well together. One example is the SIG's collaboration with the IEEE 802.15 working group, which was formed in the spring of 1999 to develop standards for wireless personal area networks. In the summer of 1999, the SIG proposed the version 1.0 Bluetooth specification, which had just been published, as a potential IEEE 802.15 standard. A task group within the 802.15 working group was then formed to draft an IEEE 802.15 standard based upon the group of Bluetooth transport protocols[2].

Extensions and Enhancements Working Groups

All of the working groups discussed in this section had their genesis in usage cases that were addressed to some extent during the version 1.0 specification development. In each case, some preliminary work was done within the SIG during its early days, but for various reasons the complete profiles for these usage scenarios were deferred. Because the SIG fully expected to complete these profiles for version 2.0 of the specification, the foundation for each was laid in version 1.0. Some of the resulting profiles will trace back to usage cases described in Chapter 3 for which no version 1.0 profile exists. The working groups dealing with version 1.0 extensions and enhancements are:

- **Personal Area Networking (PAN)**: The PAN working group is co-chaired by Microsoft and Intel and is focused on general IP networking issues, including security. As described in Chapter 15 and elsewhere, the version 1.0 specification does not define a general solution for ad hoc IP networking; it addresses only dial-up networking and LAN access using PPP. The SIG's original networking working group had some preliminary discussions of a general IP networking solution but realized that a comprehensive profile would require more time than was available for the version 1.0 specification. The PAN group was formed to continue this work, with the deliverable of a profile that addresses secure ad hoc networking to support usage mod-

2. The IEEE 802 standards are concerned only with the two lowest communication layers, the physical and data link layers. As such, only the group of Bluetooth transport protocols is relevant to an 802 standard, and this subset of the protocol stack is the basis for the 802.15 proposal.

els such as collaborative applications (much as described in "Ad Hoc Networking" in Chapter 3).

- **Human Interface Devices (HID)**: Chapter 3 described the cordless computer usage model and noted that no profile existed for it in version 1.0. The HID[3] working group is intended to focus primarily on such a usage scenario. HID refers to computer peripherals such as keyboards, mice, joysticks and the like. HID is an existing specification for the use of such devices with computers, and the HID working group, chaired by Microsoft, is charged with the development of a profile to realize the use of HID over Bluetooth links to realize the cordless computer usage model.
- **Printing**: While none of the initial usage models dealt directly with printing, it is such a common task that it was discussed in numerous working groups. The cordless computer usage model describes printer peripherals, and printing is a common example for service discovery scenarios. Co-chaired by Hewlett-Packard® (an associate member of the SIG) and Ericsson and populated by numerous printing experts, the printing working group addresses various usage cases that involve printing over Bluetooth links. These include direct-to-printer scenarios using peer-to-peer communication to print from various devices to a Bluetooth printer.
- **Still Image:** In "The Instant Postcard" section of Chapter 3, we note that that scenario is unique among the version 1.0 usage models in that it includes a personal device other than a mobile phone or computing platform, namely a digital camera. The still image working group, chaired by Nokia, works on details of image handling and manipulation in Bluetooth environments, with the instant postcard usage model at the heart of its focus area. This working group formalizes the model of image transfer as described in the instant postcard scenario, and also addresses manipulating the image, perhaps for display or printing (and thus works with the printing working group).
- **Extended Service Discovery Profiles (ESDP):** Chapter 8 described a design objective of SDP to permit coexistence with other industry service discovery protocols. The ESDP working group, co-chaired by Microsoft and 3Com, focuses on formal specifications, in the form of profiles, for mapping selected service discovery protocols to Bluetooth environments. The initial

3. Not to be confused with the hidden computing usage model.

focus for these profiles is on the Universal Plug and Play and Salutation technologies (the latter being an outgrowth of [Miller99], which described a Salutation mapping in informal terms), although other profiles for other technologies may come later.

New Applications Working Groups

While each of the working groups noted in the preceding section will produce profiles or other documentation to enable new applications of Bluetooth technology, they all have grown out of work that was started during version 1.0 specification development. The working groups discussed in this section, on the contrary, are truly new domains for Bluetooth wireless communication. While these applications might have been discussed in passing in the original SIG, there was no specific work done to enable them. Each deals with an application of Bluetooth technology that is more than just an evolutionary extension of the version 1.0 usage models. The working groups dealing with new applications are:

- **Car Profile**: Automotive manufacturers have expressed great interest in Bluetooth technology for in-vehicle communication. Chaired by Nokia, the car profile working group is investigating solutions for wireless communication within vehicles, including accessing devices and services in a car using Bluetooth links (perhaps automotive information and entertainment systems) and the use of personal devices like pagers, mobile phones and mobile computers in automotive environments. There are many possibilities for the use of Bluetooth wireless communication in vehicles. Consider scenarios such as automatically configuring personalized settings in the automobile (ventilation, seat and mirror positions, radio settings and so on) based upon personal identity carried on a Bluetooth device, or retrieving e-mail through a cellular link between the car's Bluetooth network and a larger network and then having that mail read, using voice technology, over the car's audio system. Numerous other applications can be imagined, and the car profile working group is chartered to specify many of these.
- **Richer Audio/Video (AV)**: This working group, co-chaired by Philips® and Sony® (both SIG associate members), addresses the use of Bluetooth links for exchanging audio information beyond the simple voice-quality audio specified in version 1.0, as well as motion video data. With multimedia capability on Bluetooth devices, new usage scenarios for movies and video clips,

music (with wireless headphones), video conferencing, dictation and others could be enabled. The AV working group deals with the challenges of handling this kind of data-intensive, time-critical information in Bluetooth environments.

- **Local Positioning**: Co-chaired by Microsoft and Nokia, this working group investigates the use of Bluetooth wireless technology as a means to determine the geographic location of a device (and often, then, the user of that device). Through judicious use of the properties of the short-range RF interface, Bluetooth technology can be employed to determine local (in-building) position information. The local positioning working group is chartered to provide a scheme to gather such information and make it available in a standard way to applications. The applications could then use this information for a multitude of purposes that might include selection of the "best" device to connect to in a local area, based upon proximity, locating a lost device, and so on.

Creating Additional Profiles

The foregoing are some of the working groups initially chartered by the SIG in 2000. Over time, new working groups may develop more new extensions and profiles for Bluetooth applications. The SIG's new organization promotes participation by a wider group of contributors and enables the formation of new working groups when sufficient industry interest exists for a given topic. The SIG has developed a formal process for the creation of working groups and profiles. As it does for the products discussed below, tremendous opportunity exists for innovation resulting in new applications and profiles.

BLUETOOTH PRODUCTS

This section discusses the product landscape for Bluetooth technology. We do not cite specific products or companies;[4] rather, we describe general product classes. We survey, in general terms, the first Bluetooth products to reach the market in 2000 and predict the kinds of products expected to appear over time.

4. The official Bluetooth SIG web site, http://www.bluetooth.com, includes a section that links to currently available products.

Silicon and Developers Kits

The basis of the specification is the radio and baseband; so, too, are these components the fundamental elements of products. Manufacturers building end-user products need to start with a chip set that includes the RF componentry and digital subsystems for the baseband firmware and its associated memory. Original equipment manufacturers (OEMs) have a choice among several suppliers of Bluetooth hardware. In 2000, at least seven vendors were supplying Bluetooth radio modules to the marketplace.

Most silicon manufacturers also can supply complete developers kits to accompany the radio module hardware. Developers kits typically include a circuit board with multiple interfaces to a host, along with protocol stack software that executes on the host. These developers kits allow product manufacturers to create their own products, both hardware and software, and to test and debug those products using the kits' accessible features. These kits typically allow for frequent and easily made changes to the product under development, so that the development process is expedited. Often a large portion of the product development process can be completed using the developers kits and other development tools, without having to create a final product image until late in the cycle.

In addition to general development systems, a specific class of developers tools for Bluetooth wireless technology also emerged in 2000: *protocol analyzers.* These are tools that can capture the air-interface traffic over Bluetooth links and present this information to the developer in an easy-to-comprehend fashion. These wireless protocol analyzers are analogous to their wired counterparts but do not require a physical connection to the products being tested, since they just need to intercept RF traffic. They tend to be passive receivers which capture the packets in a piconet and transfer this data to a host for processing. The processing might include separating packets from each of the various layers in the stack and displaying that data in human-readable form on the host. Protocol analyzers can be especially helpful in Bluetooth environments because the actual bit streams transferred over the air-interface can be quite complex.[5]

5. Consider all of the operations that might occur on the data before it is transmitted over the air-interface: packetization, FEC, whitening, encryption and other transformations could make the over-the-air data bear little resemblance to the logical PDUs generated by upper layers of the stack.

Since silicon, hardware, firmware and developers kits enable the production of end-user products, these were the first Bluetooth products available. Numerous such hardware and development platforms became available beginning in mid-2000.

Legacy Product Enablers

One way to quickly introduce Bluetooth technology to the marketplace is to produce "add-on" components that attach to existing products to enable them for Bluetooth wireless communication. Examples of such products include PC cards (also known as PCMCIA cards) for notebook computers and similar devices and hardware "plug-ins" (sometimes informally called "dongles"; we use the two terms interchangeably) that attach to a standard interface such as a serial or USB port.

The first PC cards with associated protocol stack software and drivers for popular PC platforms were announced in 2000, with some being demonstrated at developers conferences. These Bluetooth PC cards work similarly to other PC cards; the card is installed in the computer, along with its associated software, and the system is presented with a new interface (in this case, one for Bluetooth wireless communication). A primary advantage of PC cards here is in adding capability for Bluetooth wireless communication to existing machines in a straightforward manner. Without the purchase of a new computer, a new feature becomes available. One disadvantage is that the Bluetooth technology is not seamlessly integrated into the system, as it would be if included in the base manufactured unit. Thus performance may not be optimal, due to considerations such as antenna placement (which is necessarily on the PC card itself, or perhaps elsewhere via a cable connection; in either of these cases, fragility is one concern). In addition, PC card slots on mobile computers typically are limited to one or two, so a Bluetooth PC card occupies a slot that might otherwise be used for other features.

Interface add-ons provide a similar way to enable existing devices with Bluetooth wireless communication. These devices typically plug into an existing standard interface, such as an RS-232 serial port or a USB port. As with PC cards, a dongle with the appropriate protocol stack and driver software can enable the existing interface to support Bluetooth wireless communication. From the system's point of view, the traffic is directed over the existing port just as it would be in a cabled environment, but the interface add-on then receives and transmits that data over the Bluetooth air-interface. The advantages and disadvantages of dongles are similar to those of PC cards: they can enable immediate use of the Bluetooth technology on existing devices, but they might exhibit nonoptimal performance and fragility while taking up one of the system's interface ports.[6] Interface add-ons can be constructed for many types of devices, not just computers, thus (at least theoretically) enabling any device that exports a standard interface to make use of Bluetooth wireless communication. Handheld computers, digital cameras, printers, scanners and other devices are all candidates for add-on Bluetooth solutions. However, packaging dongles for use on small handheld devices might in some cases make the resulting device significantly larger and less convenient to use. Dongles for use with equipment that typically is stationary, such as printers, scanners and similar devices, though, is potentially quite valuable.

Because PC cards and interface add-ons can enable legacy devices to immediately utilize Bluetooth technology, these devices were some of the first end-user products to appear in 2000. The use of standard interfaces, like serial and USB ports, combined with the freedom from extensive electromechanical design and packaging as is required for integrated solutions, makes the production of these legacy device enabling products relatively straightforward.

Computers and Mobile Phones

Given the composition of the original SIG's promoter members, who have significant business interests in mobile phones and personal computers (both desktop and mobile), it is not surprising that these devices were among the first end-user products to have Bluetooth technology integrated into them. All of the version 1.0 cable-replacement usage models involve a phone, a computing platform, or both.

Among the first products to be announced and demonstrated were Bluetooth mobile phones and headsets. Most major mobile phone manufacturers indicated that they will ship handsets (and in many cases, also headsets) with Bluetooth technology in 2000. The popularity of mobile telephones results in very high volume manufacturing (in the hundreds of millions of units) of these devices; hence mobile phones are a significant influence on Bluetooth module proliferation. If a significant portion of mobile phones include Bluetooth technology, then

6. For a USB interface, this latter consideration is minimized through USB's "multi-drop" device attachment scheme.

hardware costs can decrease as manufacturing volumes increase. Achieving lower-cost Bluetooth modules then enables their incorporation into other cost-sensitive devices such as consumer electronics (discussed below).

Computers are a key device segment for Bluetooth technology. Mobile computers, especially, have a high affinity for Bluetooth wireless communication and are included in several of the version 1.0 profiles. A number of major mobile computer manufacturers planned to incorporate Bluetooth technology in their products in 2000. Furthermore, through the use of PC cards (described above), the large installed base of mobile PCs can be enabled with Bluetooth technology in the short term, in addition to integrated solutions that may be offered by computer manufacturers. Moreover, Bluetooth wireless communication is not just for PCs; prototype solutions for several handheld computers were demonstrated at developers conferences in 2000, so these devices are also expected to incorporate Bluetooth technology.

Other Products

The initial marketplace for Bluetooth wireless communication is populated by mobile telephones and computers and associated accessories and add-on components for these devices. This is not surprising, given the composition of the SIG's promoter group. But the complete SIG membership also includes manufacturers and software developers from many industries.

Given the SIG's post-version 1.0 work on printing, still image and automotive solutions, we expect to see Bluetooth technology incorporated in printers, digital cameras and automobiles in the foreseeable future. Leading manufacturers in each of these industries are actively participating in the SIG, and there is momentum to produce equipment with Bluetooth wireless communication capability.

As the technology's costs continue to decrease over time, many consumer electronic devices may incorporate Bluetooth wireless communication. Again, with leading industry players being involved in the SIG, Bluetooth devices including televisions, stereos and other audiovisual devices are expected to appear in the marketplace. Chapter 3 discussed what the SIG calls "the ultimate headset," which can be used with computers and mobile telephones. As other audio devices incorporate Bluetooth technology, such a headset might be used with portable CD players, automobiles, stereos and other such devices, perhaps enabling an "ultimate ultimate headset." Other products in which Bluedata over the Bluetooth air-interface. The advantages and disadvantages of dongles are similar to those of PC cards: they can enable immediate use of the Bluetooth technology on existing devices, but they might exhibit nonoptimal performance and fragility while taking up one of the system's interface ports.[6] Interface add-ons can be constructed for many types of devices, not just computers, thus (at least theoretically) enabling any device that exports a standard interface to make use of Bluetooth wireless communication. Handheld computers, digital cameras, printers, scanners and other devices are all candidates for add-on Bluetooth solutions. However, packaging dongles for use on small handheld devices might in some cases make the resulting device significantly larger and less convenient to use. Dongles for use with equipment that typically is stationary, such as printers, scanners and similar devices, though, is potentially quite valuable.

Because PC cards and interface add-ons can enable legacy devices to immediately utilize Bluetooth technology, these devices were some of the first end-user products to appear in 2000. The use of standard interfaces, like serial and USB ports, combined with the freedom from extensive electromechanical design and packaging as is required for integrated solutions, makes the production of these legacy device enabling products relatively straightforward.

Computers and Mobile Phones

Given the composition of the original SIG's promoter members, who have significant business interests in mobile phones and personal computers (both desktop and mobile), it is not surprising that these devices were among the first end-user products to have Bluetooth technology integrated into them. All of the version 1.0 cable-replacement usage models involve a phone, a computing platform, or both.

Among the first products to be announced and demonstrated were Bluetooth mobile phones and headsets. Most major mobile phone manufacturers indicated that they will ship handsets (and in many cases, also headsets) with Bluetooth technology in 2000. The popularity of mobile telephones results in very high volume manufacturing (in the hundreds of millions of units) of these devices; hence mobile phones are a significant influence on Bluetooth module proliferation. If a significant portion of mobile phones include Bluetooth technology, then

6. For a USB interface, this latter consideration is minimized through USB's "multi-drop" device attachment scheme.

hardware costs can decrease as manufacturing volumes increase. Achieving lower-cost Bluetooth modules then enables their incorporation into other cost-sensitive devices such as consumer electronics (discussed below).

Computers are a key device segment for Bluetooth technology. Mobile computers, especially, have a high affinity for Bluetooth wireless communication and are included in several of the version 1.0 profiles. A number of major mobile computer manufacturers planned to incorporate Bluetooth technology in their products in 2000. Furthermore, through the use of PC cards (described above), the large installed base of mobile PCs can be enabled with Bluetooth technology in the short term, in addition to integrated solutions that may be offered by computer manufacturers. Moreover, Bluetooth wireless communication is not just for PCs; prototype solutions for several handheld computers were demonstrated at developers conferences in 2000, so these devices are also expected to incorporate Bluetooth technology.

Other Products

The initial marketplace for Bluetooth wireless communication is populated by mobile telephones and computers and associated accessories and add-on components for these devices. This is not surprising, given the composition of the SIG's promoter group. But the complete SIG membership also includes manufacturers and software developers from many industries.

Given the SIG's post-version 1.0 work on printing, still image and automotive solutions, we expect to see Bluetooth technology incorporated in printers, digital cameras and automobiles in the foreseeable future. Leading manufacturers in each of these industries are actively participating in the SIG, and there is momentum to produce equipment with Bluetooth wireless communication capability.

As the technology's costs continue to decrease over time, many consumer electronic devices may incorporate Bluetooth wireless communication. Again, with leading industry players being involved in the SIG, Bluetooth devices including televisions, stereos and other audiovisual devices are expected to appear in the marketplace. Chapter 3 discussed what the SIG calls "the ultimate headset," which can be used with computers and mobile telephones. As other audio devices incorporate Bluetooth technology, such a headset might be used with portable CD players, automobiles, stereos and other such devices, perhaps enabling an "ultimate ultimate headset." Other products in which Bluetooth technology could be used include universal remote controls, household appliances and even toys. Bluetooth technology certainly has the potential to be widely deployed in enterprises, homes and public venues. Successful introduction of the first devices can help to enable positive perception, user experience and value for users of many types of devices. We believe that Bluetooth wireless communication is well positioned to make inroads in many different devices and marketplaces.

17

Concluding Thoughts

In this book we have presented many facets of Bluetooth wireless communication. Part 1 introduced the technology, discussed the origins of the SIG and presented an overview of wireless communication concepts leading to the development of the Bluetooth technology and specification. Parts 2 and 3 delved into the specification, aiming to make it more accessible and easily understood. Our choice of the title for this book is based upon our endeavor to reveal the specification's background and development in addition to interpreting its contents. Part 2 covered the core specification, or volume 1, focusing on the protocol stack. Part 3 addressed volume 2 of the specification, the profiles.

We conclude with some forward-looking remarks about the directions in which Bluetooth wireless communication is likely to proceed in the future.

Interoperability

The motivation for producing profiles–over 400 pages in version 1.0, with more profiles continuing to be developed–is to foster interoperability. Indeed, the formation of the SIG itself was aimed at developing an open specification with backing from leaders in the computing and telecommunications industries. The SIG is well aware that many promising new technologies have not gained market traction for various reasons, and the SIG believes, as we do, that interoperability is a key attribute that can enable the initial and continued success of Bluetooth technology.

Besides expending tremendous effort to develop the profiles, the SIG also sponsors other activities geared toward fostering interoperability. Events called *unplugfests* are held from time to time. During unplugfests, vendors can test their Bluetooth solutions with those of others to determine how well the different implementations interoperate. Unplugfests are informal gatherings where developers, under nondisclosure agreements and within a spirit of cooperation, can judge the precision and completeness of their protocol stack implementations. These events have been popular and have allowed developers to discuss and test their implementations with each other, helping to resolve conflicts and work toward producing robust, interoperable solutions.

The compliance testing and logo programs provide more formal methods for testing and certifying Bluetooth implementations. The certification program is not covered in detail in this book; a detailed presentation could perhaps be a book unto itself. The compliance testing and logo programs were still maturing in 2000 and are likely to continue to evolve over time. In general, these programs revolve around a process for formal testing of a Bluetooth implementation by a SIG-certified test body. The types of tests vary with the type of implementation being tested. Once a product is certified through the testing process, the product can sport the Bluetooth logo (depicted in Chapter 1) as an indication of its compliance with the specification. The SIG publishes rules for the use of the logo; these rules and other authoritative information about compliance testing can be found at the official Bluetooth web site, http://www.bluetooth.com.

OPPORTUNITIES

Innovation is the lifeblood of the computing and communications industries. The Bluetooth technology fosters innovation and presents many opportunities to many people. First and foremost is the value it can provide to end users in the form of convenience and new applications of the technology. As the previous chapter pointed out, product opportunities abound for manufacturers and software developers. Like most new technologies, Bluetooth wireless communication presents opportunities for education and consulting by those who choose to become immersed in the technology. Indeed, this book and others on the topic are intended to educate and widely disseminate information about this exciting new technology. This book has not covered every facet of Bluetooth wireless communication–besides the new subjects which will arise as the technology evolves, there is still room for investigating other topics in more depth than we have done here. Testing and certification, WAP interoperability, software design and development, silicon and antenna design and development and other subjects are all ripe for further exploration.

With a solid and robust foundation, exceptional industry backing, a detailed specification, tremendous momentum, and dedicated product developers worldwide, Bluetooth wireless communication is poised to become a major influence in high-technology industries. Bluetooth technology has made great strides since its inception, and we believe that King Harald would be proud of his namesake.