Enabling Rapid Wireless System Composition through Layer-2 Discovery

Shivani Sud, Roy Want, Trevor Pering, Barbara Rosario, and Kent Lyons, Intel Research

Abstract

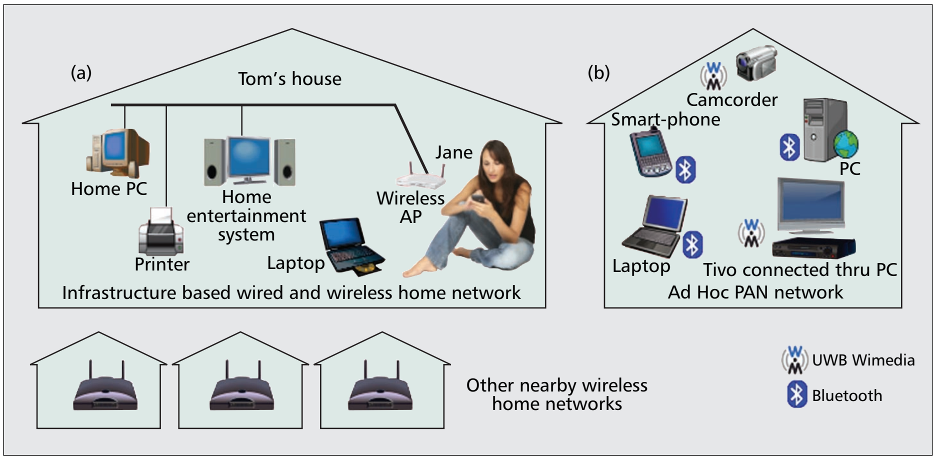
Although small mobile computers have processors whose capabilities are increasing, often they still are resource constrained in terms of performing many common computing tasks. Composition is a technique that overcomes this limitation by wirelessly connecting several nearby devices together to share their resources to create a logical platform that is better suited to a given task. However, current standards for ad hoc wireless networks were not designed for this goal, and currently, wireless computers cannot discover services on other devices without first making layer-3 network connections between them. This requires users to engage in a lengthy and repetitive sequential connection process to find these services on the various computers and is a barrier for effective use. In this article, we quantify the layer-3 discovery overhead for both WiFi and Bluetooth and propose a mechanism to address this problem enabling users to rapidly form ad hoc compositions using a layer-2 service discovery mechanism. To achieve this result, we propose extensions to existing wireless standards by adding service information to beacons that typically are used for device discovery. An architecture is described that also enables legacy service information to be aggregated with additional service descriptions specific to composition and further encoded in layer-2 wireless beacons. Finally, we present measurements of layer-2 discovery implemented using the nascent UWB wireless standard, demonstrating its efficacy for composable systems.

The commercialization of modern consumer electronics, personal computing, and networking products was developed successfully around a component model, enabling users to incrementally buy devices and connect them together to build engaging and complex systems customized for their applications. Entertainment centers are a typical example, enabling systems to be incrementally built from amplifiers, DVD players, TVs, and so on, using analog cabling. Personal computing has enabled similar multi-device systems to be built from various computers, printers, and network storage devices, all of which now also are capable of supporting rich digital-media applications, enabling a *Digital Home* computing and entertainment environment. However, most of these systems were designed around wired networks using protocols that provide standardization for connecting systems together but are difficult to use when dynamically composing systems from diverse sets of devices on-the-fly. Now that wireless standards such as WiFi [1] and Bluetooth [2] (and recently introduced ultra-wideband [UWB] [3]) are used widely as networking media, we no longer have the inconvenience of physically plugging components together and instead enjoy the flexibility to wirelessly discover nearby resources, dynamically connect to them [4], and potentially reconfigure a multi-component system at the touch of a button [5] in a process we call *composition*. However, although this potential exists, composition currently is not a simple task, because a lack of wires actually can make it *more* difficult to choose a set of components and specify how they should be interconnected. Unfortunately, existing protocols used for discovery [6] in wired networks do not solve this problem for composition, and they cannot be used for wireless service discovery to make dynamic connections without being part of a much longer and involved process for the user.

To illustrate the deficiency of the wireless discovery processes based on existing standards, consider the following scenario in which Jane, a mobile user, takes her wireless smartphone to Tom's house, a modern digital home. While Tom is out running an errand, Jane would like to connect her smartphone to Tom's wireless multimedia service, which contains his CD collection. Using wireless local area network (WLAN) discovery (infrastructure mode), she finds several wireless networks that her mobile can connect to, one of which is Tom's, and others that are in nearby houses (Fig. 1a). Unfortunately, it is not clear which network is Tom's. Although she is looking for a media server, the wireless discovery process tells her only about the service set identifiers (SSIDs) of the access points and therefore, provides no indication on which network the service is running. To find the intended service, Jane must connect to each of the discovered access points in turn and for each one, see if her mp3 player application has discovered the correct service. Furthermore, if the wireless networks are secured, then she must attempt to authenticate to each network in turn, dramatically increasing the connection overhead.

If this scenario had been constructed around Bluetooth using the personal area network (PAN) profile, WiFi in ad hoc mode, or Wireless Link Protocol (IP over UWB), then

0890-8044/08/$25.00 © 2008 IEEE



■ Figure 1. *Example of a user trying to find a specific wireless media service: a) Jane finds several access points for networks attached with services available; b) there are several Bluetooth and UWB devices also offering services. In either case, a user will still need to make connections using trial and error to locate the media service.*

A. Device discovery	B. Device selection	C. Layer 2 connection	D. Layer 3 connection	E. Layer 3 service discovery	F. Service selection
(Inquiry/scan channels)	(user choice or pre-configured)	(media dependent)	(IP address through DHCP/static/Auto IP)	(UPnP/ZeroConfig)	(user choice or pre-configured)

■ Figure 2. *Stages in wireless service discovery to create dynamic compositions.*

similarly, Jane must connect to each computer that she discovers until she finds the desired media service (Fig. 1b). In this case, the search time would be proportional to the number of nearby computers, potentially making this a long and tedious process for Jane and a serious impediment for all mobile users in a world with ever-increasing numbers of wireless devices. Again, authentication is problematic; however, she could authenticate directly with that device (perhaps using an access code displayed on its screen) when she does find the media server. The ad hoc approach has the advantage of allowing her access to just the one device, instead of necessitating access to Tom's entire network.

This article provides a solution to Jane's problem by incorporating high-level service descriptions in the link-layer discovery protocol of a wireless personal area network (WPAN), thus enabling a user to learn about services and the machines they reside on, prior to any wireless connections being made. This solution avoids the wireless local area network (WLAN) network-access problem described previously, and reduces the WPAN discovery overhead, reducing the time required to establish both simple wireless connections and complex compositions. We first frame the problem by providing a detailed overview of the issues surrounding the wireless discovery process and quantify the deficiency of existing wireless discovery mechanisms. Then, we describe a specific solution that utilizes layer-2 beacons that are available in the nascent UWB WiMedia standard to perform rapid ad hoc service discovery and finally, conclude with results comparing UWB with other radios and with lessons learned.

Composition and Service Discovery

Sharing resources between devices through wireless services is a key building block of a composition system. For example, users may want to move content between their mp3 players and their home entertainment systems; or from various recording devices (such as cameras, camcorders, or camera phones) to a home PC. Existing network file system protocols such as Samba can provide this capability. Users also might wish to make use of a larger display than is available on their mobile device. Service solutions such as virtual network computing (VNC) and Remote Desktop Protocol (RDP) are available for this purpose. In fact, sharing resources in general can be achieved by using existing network services or designing new ones; and all these services typically are advertised at the IP layer — layer-3 in the open systems interconnection (OSI) model. However, in existing wireless standards, there are no mechanisms to advertise these services before a link-layer (layer-2) connection is established. Thus, layer-2 service advertisement and discovery is an essential ingredient of a wireless system that supports dynamic composition.

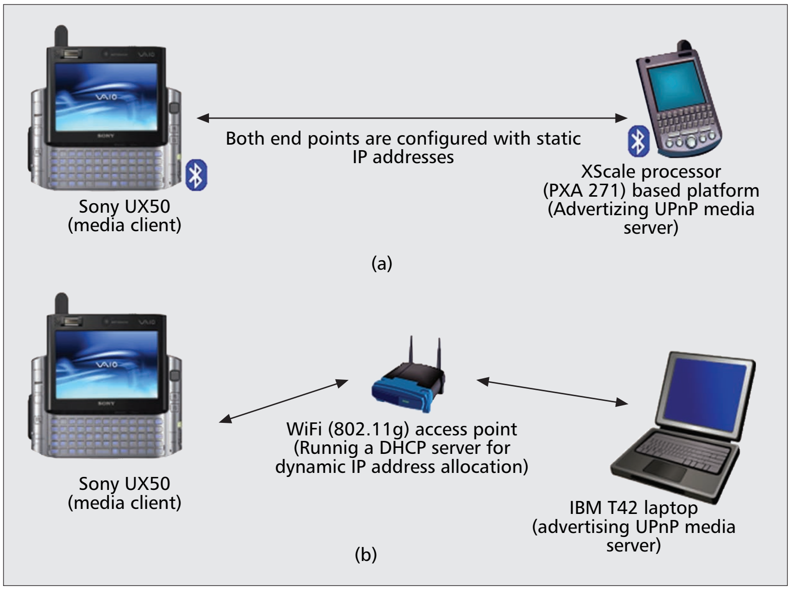
Service Discovery in Infrastructure-Based WLANs

In most managed networks (wired and wireless), peer-to-peer discovery is achieved through universal plug and play (UPnP) [7] and various zero configuration networking (ZeroConf) [8] solutions. These are IP-based (layer-3) protocols that allow for pervasive discovery of services and devices. IP addresses in the peer-to-peer network are assigned either through a Dynamic Host Configuration Protocol (DHCP) server or dynamically using IPv4 or IPv6 link local addresses (RFC 3927 and RFC 2462). Service discovery in UPnP is performed using Simple Service Discovery Protocol (SSDP), resulting in multicast discovery messages to control points (entities that listen for new services as they become available on the network). When other devices on the network receive these multicast messages, the embedded service information alerts them and provides enough information to contact the service.

To use any of these protocols that are defined at the IP layer or above, network devices must have a prior link-layer connection that enables them to listen to the multicast advertisements from peer devices. In the case of wired network devices, this is an implicit process because the link-layer connection is established by the action of plugging them into a local area network.

Service Discovery in Ad Hoc WPANs

The relatively recent ubiquity of wireless mobile devices provides us with an opportunity to design systems that can be connected in an ad hoc fashion and to build logical platforms that are better suited to user tasks than the individual devices, all without requiring knowledge of the component device capabilities in advance. In existing ad hoc networks, each device must connect to others in a five stage process (A through E), shown in Fig. 2. Only at stage F can the user choose the service required. Each repetition of this process adds to both the time and power consumption of the discovery process and becomes a significant cost when establishing multiple connections before actually discovering a peer that offers the services it requires. Also, the process of establishing multiple link-layer connections incurs an additional cost when the requesting device first must authenticate with each of the discovered peer devices in the proximity.



■ Figure 3. *Showing experimental configuration for a) Bluetooth; b) WiFi.*

Layer-2 Service Discovery for WPANs

The layer-2 service announcement is a technique for advertising service information incorporated in device advertisements and is designed to overcome the existing limitations of layer-3 service discovery in an ad hoc wireless network. It enables a device looking for a particular service to gather sufficient information about all the peer devices that are advertising services before it establishes any link-layer connections. This information may include the service names and resources being offered and where and how they can be accessed, all as an implicit part of the wireless discovery process. One of the following two generalized approaches can be used for achieving this goal:

- Beacons
- Probes

Beacon Model for Service Discovery — Beacons are periodic transmissions from wireless devices used to advertise the presence of a device and to establish its transmission schedule with peers. In modern wireless standards such as UWB and 802.11n, application-specific data can be encapsulated in fields called application-specific information elements (ASIEs) that are transmitted as a part of the beacon information. These ASIEs, although constrained in size, can be used to transmit summaries of the services available on a device. In all likelihood, the full service and resource advertisement information will exceed the maximum transmission size allowed for a single beacon. To overcome this problem, it is possible to spread the service information over multiple beacons that are being issued repeatedly from these devices. In this way, a nearby wireless device can listen for the fragmented service information and collect the components over time to reconstruct the full description. If any parts of the sequence are missing, the device can wait for the next occurrence in subsequent beacons. Because the available bandwidth in UWB is high, and beacons consume relatively small amounts of bandwidth, this process does not adversely affect the utilization of the network. Also, a beaconing device can set itself in a low-power state in between beacons, listening for connection requests only for a short period after each beacon it sends, which is an advantage for power constrained mobile devices.

Probe Model for Service Discovery — An alternate approach for discovering services is to use probe requests. When a device joins a PAN, it sends out multicast probe requests to other devices in the PAN soliciting their shared services. Peer devices can unicast their response to the requesting device (reducing the chatter on the link layer). They also could include their own service advertisements as a part of the probe response. This mode of operation could reduce significantly the amount of beacon traffic generated by each device; however, because the probe requests can occur at any time, each of the peer devices in the vicinity must be in an active power state to be able to listen to the probes and respond to them. This has adverse power implications for small mobile devices with constrained battery lifetimes.

Properties of Radio Technologies Supporting WPAN

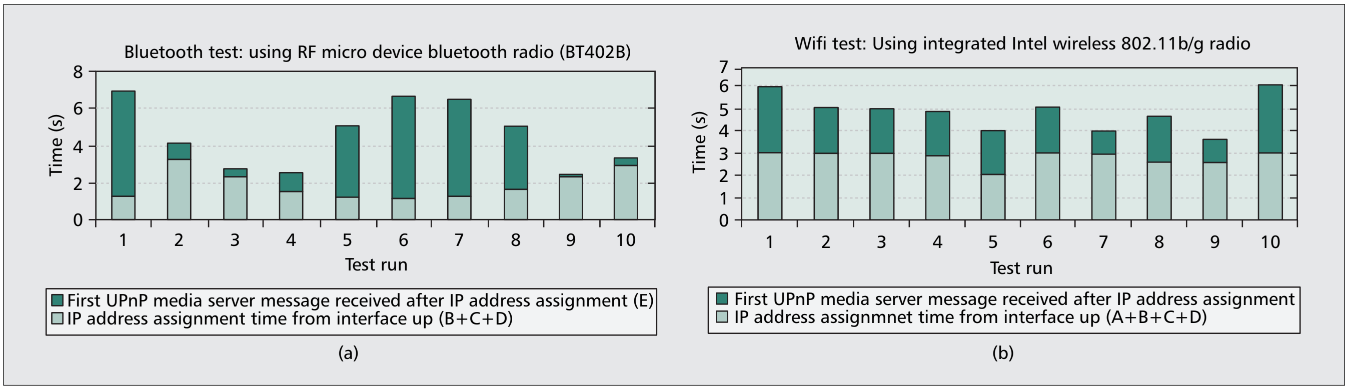
There are many radio technologies that support localized wireless networks including Bluetooth, WiFi (in ad hoc mode), and UWB. Each of these radio technologies emerged from different segments of the computing industry and therefore, have different constraints and characteristics as described in the following.

Bluetooth is a radio technology that operates over a short-range, typically within ten meters and with a transmit power of 1 mW. When a device is not wirelessly connected, it can transition to a sleep-state and still be discovered while consuming much less power than in its active state. Originally, the standard was designed to target the cell phone industry, allowing communication to peripherals such as headsets and printers and data exchange with nearby PCs. At any given time, data can be transferred between a master and up to seven active slave devices. However, to maintain flexibility, the devices can switch roles and at any time, a slave can become the master. The PAN profile allows devices to be connected together in a PAN supporting standard TCP/IP networking over the Bluetooth link-layer protocol.

WiFi 802.11n is an emerging standard that provides for maximum data rates up to 600 Mb/s (typical data rates are around 200 Mb/s) with a range of 70 meters. It provides significantly faster throughput, reliability, and range compared to earlier 802.11 standards and is touted to be the replacement technology for wired networks in the home and office. An 802.11n device has a nominal radiated transmit power of about 50 mW. 802.11s is an accompanying standard that provides for mesh-networking connections that would enable ad hoc services without an access point.

UWB is a radio technology that provides very high data-transfer rates (480Mb/s) over ten meters with a power consumption that can be tuned to the bandwidth requirements of an application, and these characteristics make it an ideal radio technology for WPAN applications such as streaming video or sharing storage in home or office environments. Its power management features also make it extremely desirable for mobile small form-factor devices where extending battery life is an essential requirement. Current UWB prototype radios can operate with a radiated transmit power as low as 0.1mW.

802.11n has an average radiated transmit power of about 500 times the average power emitted by an UWB radio for a range of 10 m; this makes UWB a more suitable candidate for the WPANs [9] that involve mobile devices. Bluetooth, although low-power by this measure, does not support enough bandwidth for many multi-media applications and in fact, consumes more *energy per bit* to transfer a file than the other



■ Figure 4. *Graph showing UPnP total time for media service discovery in PANs based on WiFi (802.11g) and Bluetooth showing large variance in service discovery times.*

radios that consume more average power, mainly because it takes far longer to perform the task.

Measurements from Established Wireless Standards

We measured the service discovery times for a UPnP advertised media service on two separate wireless networks: one using a Bluetooth radio and another using the WiFi 802.11g standard to demonstrate the typical latencies a user could experience when using the wireless products that are available today (Fig. 3).

For each experiment, the set up consisted of a media player client running on a wireless mobile device and another wireless device advertising a UPnP-based media service; and tests were repeated over ten iterations for each interface. The client used for all tests was a Sony UX50, an ultra-mobile PC (UMPC) running the WinXP operating system and based on Intel's Core Solo U1300 processor clocking at 1.06GHz. The media client was a Digital Living Network Alliance (DLNA) v1.5 compliant media player, and the media server was set up using Intel's publicly available UPnP Media AV Server.

- **Bluetooth**: in the first test using the Bluetooth radio standard, the UPnP Media Server ran on a 400 MHz XScale processor (PXA 271)-based platform, supporting a Linux 2.4.19 kernel. The client UX50 ran an application to control the IVT BlueSoleil Bluetooth stack v2.0 to discover nearby Bluetooth devices, which provided an interface for a user to establish a connection with one device from a list of others discovered nearby. Device discovery (phase A, inquiry, in Fig. 2) was performed only once before the tests and is not included in the media service discovery time. Static IP addresses were assigned to the Bluetooth interfaces on both devices.
- **WiFi**: in the second test using the 802.11g radio standard, the UPnP media server ran on an IBM T42 laptop with a 1.7 GHz processor, 1GB RAM, supporting Microsoft Windows XP. The access point (AP) also was set up as a DHCP server. The T42 ran a DHCP client that was assigned an IP address by the server. The WiFi interface is configured to automatically connect to the AP, so the device selection time (phase B in Fig. 2) includes the inquiry time (phase A).

In both tests, the Microsoft Network Monitor v3.1 was used to collect traces of wireless traffic over the corresponding radio interfaces available on the UX50 (all unused radios were shutdown during the test).

In Fig. 4, we see there is both a significant time lapse and variability between the point at which a device is detected at layer-2 and the time the services are discovered, represented by the total time shown for each test run. The average time for a device to be available at layer-3 (IP layer — lower part of the bar) was 1.5 seconds for Bluetooth and 2.8 seconds for WiFi. The average time for media service discovery (total bar height) was 4.2 seconds for Bluetooth (ranging from 3.3–6.9 seconds) and 4.8 seconds for WiFi (ranging from 3.6–6.0 seconds).

In ad hoc networks (supported by Bluetooth and emerging UWB), the device and service availability time at L-3 scales linearly with the number of devices in the WPAN, and similarly for APs in infrastructure mode, making the task of composition much slower and discouraging to users. For example, consider a scenario similar to that described in the introduction: if there were six nearby access points or devices discovered, and the user takes three seconds to think about a choice and make a selection, it could take up to $(4.2 + 3) \times 6 \sim 43$ seconds on the Bluetooth network or $(4.8 + 3) \times 6 \sim 47$ seconds on the WiFi network to browse through all of them. This would not scale well in the future when probably there will be considerably more devices in the environment.

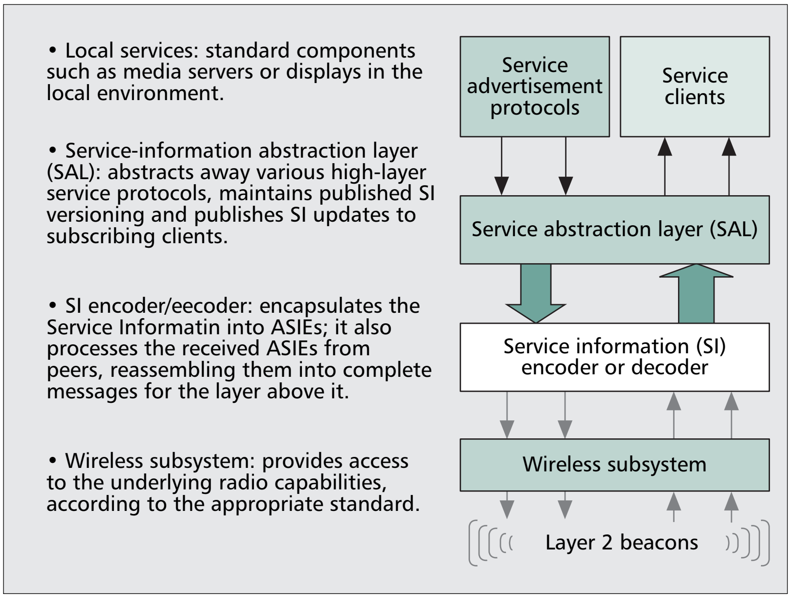
In the remainder of this article, we consider how the service discovery problem can be mitigated by using layer-2 service advertisements. Further, we provide an analysis of the overhead for service connection time using existing protocols versus the time benefits gained by modifying layer-2.

UWB Service Advertisements

We explore the implications of layer-2 service discovery using the nascent UWB standard [3] that allows for extensions to support protocol enhancement.

Introduction to UWB and WiMedia

UWB has been standardized by the European Computer Manufacturers Association (ECMA), and provides a convergence layer (WiMedia) that allows multiple application stacks, such as Wireless USB, Wireless 1394, and Wireless Link Protocol (IP over UWB) to coexist and share a single UWB radio platform. This allows for seamless reuse of traditional applications based on IP technologies without any disruption. UWB devices discover peers when they move in range of each other by using a beacon protocol that is a part of the UWB MAC, to form logical *beacon groups*. The UWB MAC layer channel format comprises of super frames, which are periodic time intervals used to coordinate frame transmission between devices. Each super frame consists of a beacon period followed by a data period. The beacon period provides information about power management and channel reservation and coordinates the beacon transmission times of all nearby devices, allowing for dynamic network organization. During the data period, devices send and receive data using priori-



- Local services: standard components such as media servers or displays in the local environment.
- Service-information abstraction layer (SAL): abstracts away various high-layer service protocols, maintains published SI versioning and publishes SI updates to subscribing clients.
- SI encoder/eecoder: encapsulates the Service Informatin into ASIEs; it also processes the received ASIEs from peers, reassembling them into complete messages for the layer above it.
- Wireless subsystem: provides access to the underlying radio capabilities, according to the appropriate standard.

Figure 5. *Architecture diagram for UWB L2 service discovery. The service information (SI) and wireless subsystem components would be replicated for another radio technology such as 802.11, but would share a common SAL.*

tized contention access (PCA) or in reservations established using the Distributed Reservation Protocol (DRP).

Calculations Supporting Layer-2 Service Announcements for UWB

The amount of data that can be broadcast in a UWB beacon is governed by the beacon length, transmission frequency, and beacon data rate. The periodic UWB super frame is 65,536 μs long and can have a maximum of 96 beacon slots (85 μs each). However, only 46 UWB devices can participate in a UWB beacon group and hence form a UWB-based PAN (the extra slots allow two beacon groups to be merged). Each device can transmit only one beacon frame per beacon period within a super frame that should not exceed 85 μs, which after subtracting a 10 μs guard-time and 12 μs inter-frame spacing time, allows for a 63 μs transmission time. The transmission rate for beacons at 53.3 Mb/s constrains the maximum beacon size to 53.3*63 = 3357 bits = 419 bytes.

Additionally, beacons can contain ASIEs that have a maximum size of 256 bytes (with 254 bytes for payload) that allows for transmitting information independent of vendor specific information. Given a 419-byte beacon, it can carry, at most, only one full-sized ASIE. The complete set of services that a device wants to advertise might far exceed this size limit; hence, we must spread the service and resource advertisements across multiple beacons that will be transmitted using multiple UWB super frames. This will require a mechanism at the listening device to reassemble the fragments and recover the original information.

Applying the parameters of the UWB protocol to the UPnP media server service discovery example in the introduction, consider that the average size of an SSDP media server message is around 400 bytes (measured from our test system); so it would take two ASIEs (and hence two beacons) to transmit the complete SSDP message. As each device transmits its own beacon with every super frame, devices can advertise their services in an interval of 65 ms*2 = 130 ms. Therefore, even if one in four of the beacons were lost, using layer-2 discovery, the relevant information would still be accessible in less than half a second. Further, to enable mobile devices to prolong battery life, a service beacon could be transmitted as infrequently as four per second and still result in a responsive sub-second service notification for the user.

Our layer-2 beacon service discovery approach uses this basic capability by encoding service information in UWB ASIEs, as described in the following section that describes the architecture.

Architecting a Layer-2 Solution

The goal that motivates the design of a layer-2 service discovery mechanism is to provide support for faster service detection in an ad hoc PAN, enabling rapid multi-device composition while reducing the power consumption of participating wireless mobile devices. The following architecture is well suited to both UWB and 802.11n and makes only incremental additions to the standards to facilitate easier industry adoption. Using this approach, it also is possible to make use of existing service discovery protocols such as UPnP by incorporating existing service descriptions into the beacon ASIEs, thus bypassing the requirement to establish an IP connection before the information can be displayed for existing services.

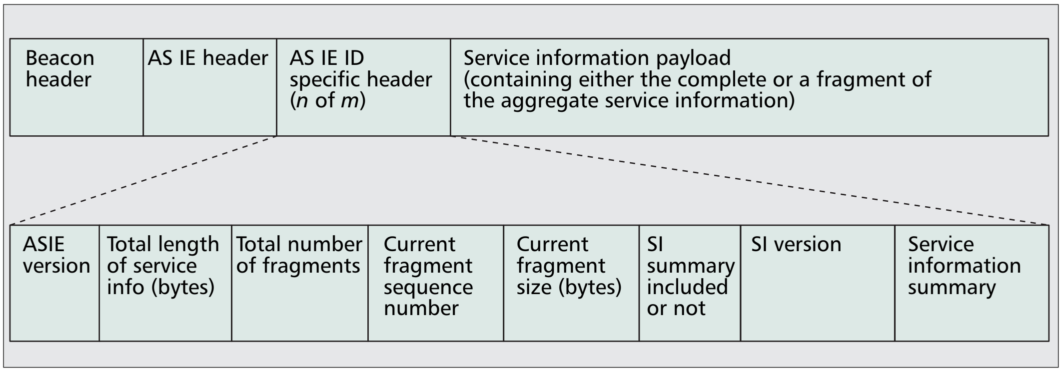
Figure 5 details the overall architecture of the layer-2 discovery mechanism. It consists of four key components: local services, a service abstraction layer (SAL), a service information (SI) encoder/decoder, and the underlying radio subsystem itself. The SAL and the SI encoder/decoder provide the core of the layer-2 discovery mechanism described in the following.

Service Abstraction Layer

The SAL layer abstracts away service discovery protocols operating at layer-3, extracting the required information from service advertisements, and encodes and aggregates the various service descriptions for transmission in the layer-2 beaconing process. At the receiving host, SAL acts as a dissemination broker, decoding the received composite descriptions and mapping them for the local clients using standard layer-3 protocols. SAL filters the service descriptions it periodically receives from peer devices using versioning information to notify recipients only when service information has changed.

For the purpose of composition, there are a small core set of services that are of high importance for creating a logical platform, allowing physical resources to be shared between computers. Examples include display and associated keyboard and mouse devices (via VNC), storage (via Samba), or broadband connections (via IP bridging). To ensure these service advertisements are received with high probability during the discovery process, a summary of these services, represented as a bitmap, is included in the ASIE of every beacon. The remainder of the ASIE can be used to represent the comprehensive service descriptions that may be fragmented further among beacons. The summary bitmap should be short enough that it remains a small percentage of the total payload capacity, so we use 32 bits (4 bytes) with one bit representing each core service, allowing an advertisement of 32 core services. This map can be standardized for a particular ASIE ID version.

In addition to the SI version and the service summary bitmap, the SAL layer optionally can include information about encoding, compression, and encryption that may be employed usefully for reducing the size of beacons or ensuring their privacy.



■ Figure 6. *Proposed ASIE format with fields to support fragmentation and reassembly of the embedded service information.*

Service Information Encoder/Decoder

The service information received through the host SAL must be packaged into a format appropriate for the underlying layer-2 radio advertisement. For UWB, encoding involves packaging the information into an ASIE or fragmenting it sequentially over multiple ASIEs across multiple (preferably sequential) beacons. The decoder module receives service advertisement fragments from the underlying MAC layer, reassembles them into the correct sequence, and then extracts the aggregate service information, ready to pass to the SAL layer.

A typical ASIE header includes an ASIE identifier, several of which are already defined by the standard, but the remainder can be assigned to vendors for their own use. We leverage a vendor-specific value for our work and define the payload for our identifier as shown in Fig. 6. Because the beaconing process is unreliable, data may be lost, and we must build in resilience when service information is fragmented across beacons. This is achieved by adding fields to track the sequence numbers of the fragments, allowing the receiving device to correctly re-sequence the fragments. As described previously, the service information version number and a service summary also are included in each beacon to enable efficient processing of repeated information, notifying clients only when services change, and ensuring a timely and compact representation for important services in every beacon.

Results: UWB Service Discovery Time at Layer 2 and 3

To validate our theoretical analysis of service discovery timing using the UWB standard, we performed experiments on a prototype ECMA-compliant UWB radio and software stack developed by an Intel product group. Our test platform was based on two notebook PCs, each with a universal serial bus (USB) 2.0 interface to the UWB radios. The client notebook that was instrumented to measure service discovery times was an Intel Pentium M processor at 1.7 GHz with 1 GB RAM, running the WinXP operating system. We modified the networking stack for the radios to support our SAL architecture. The composition framework [10] is used as an application layer service aggregator, which facilitates discovery of platform services among devices discovered nearby. We use this configuration to perform two tests: the first to validate that when using UWB and UPnP discovery at layer-3, the service discovery time is similar to the earlier results for Bluetooth and WiFi, and the second to measure the rapid service discovery possible at layer-2 using beacons.

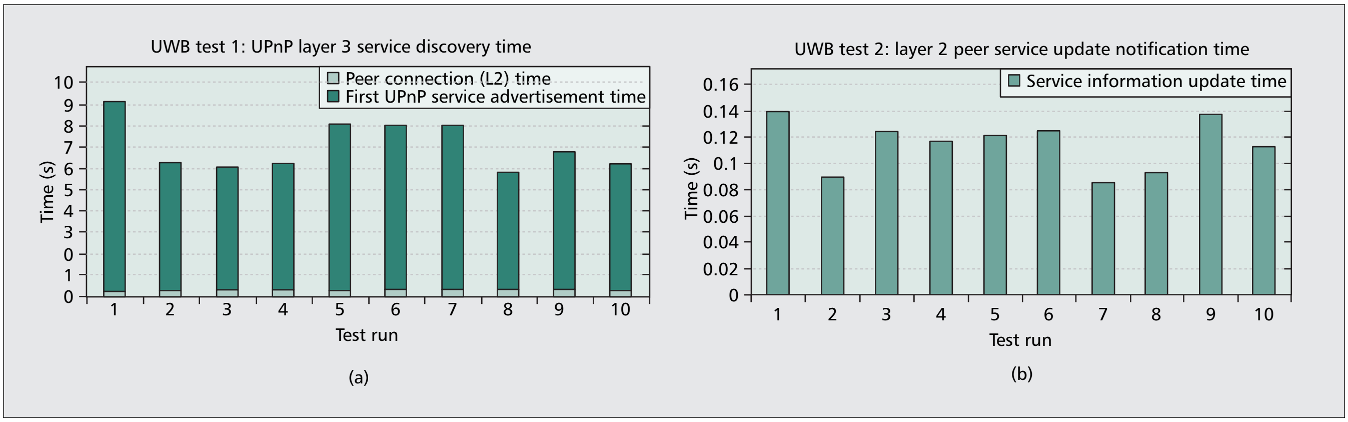
Test 1: PC #1 (server) is advertising a UPnP media service, and PC #2 (client) is running a UPnP tool called a "device sniffer" to log the UPnP service discovery protocol messages with millisecond resolution timestamps. The two UWB radios are configured to auto-connect on detecting a peer, using static IP addresses. We measure the time between when a UWB peer device is detected and a layer-2 connection is established with the peer and then, the time from that point onward to receive the first UPnP SSDP message. The total of the two is the UPnP (layer-3) discovery time. The results for ten runs are shown in Fig. 7a.

Test 2: we measure service discovery time at layer-2, with PC #1 (server) and PC #2 (client) in a quiescent state, but wirelessly receiving each other's UWB beacons, and with no network connection initiated. We then update service information advertized by PC#1 and measure the time interval between this update and when PC #2 receives the service updates. Because these two events occur on two different PCs whose clocks are not synchronized, we have added code to the client and server to flag these events as physical signals on ports attached to each PC. The two ports are connected to two channels of a Tektronix TDS 7404 oscilloscope triggered by the first event and capturing the second event as a time trace. The time difference between the two events is read manually, and the results of ten test runs provide the data for Fig. 7b.

Discussion of Results

Examining the results for UWB discovery using UPnP shown in Fig. 7a, we find the average service discovery time to be 6.8 seconds (range 5.8–8.8 seconds), which is similar in magnitude to the 4.2 seconds for Bluetooth and 4.8 seconds for WiFi in the experiments described previously. We expect that the ~2 seconds extra time measured for UWB is due to properties of the prototype adaptation layer protocol, WLP, implementation that is not optimized for product at this time.

Figure 7b shows the time required for a *change* in service discovery information with the beacon-based protocol, using the SAL architecture described in the previous section, which is on average only 0.11 seconds (range 0.08–0.14 seconds). This is a clear improvement over our previously described tests that were run for UPnP discovery (with typical system software configurations): using Bluetooth — a factor of 4.2/0.11 ~ 38x, WiFi — 4.8/0.11 ~ 44x, and our prototype UWB system — 6.8/0.11 ~ 62x. Although these numbers do not represent a fundamental comparison of the radios, and there are necessary differences between the test set ups, they are representative of the way service discovery is used with these technologies. We can summarize by saying layer-2 discovery provides a speed on the order of 50x over typical layer-3 solutions. Furthermore, if multiple services are being advertised by a variety of nearby computers, devices announcing their presence with UWB beacons can advertise their service information in the same super frame, and therefore, peer



(a)

(b)

■ Figure 7. *UWB service discovery timing: a) test 1: using UPnP at layer 3; b) test 2: using beacons at layer 2.*

devices can receive distributed service updates at the same time; for our system this was within an average of 110 ms to see any changes. Therefore, the final service selection will be governed by the user's thinking time (we used three seconds as an estimate for this earlier) and thus, will no longer be perceived as slow (e.g., the ~43-seconds example from earlier), or a source of complexity, requiring users to exhaustively create multiple wireless connections to discover all relevant services.

Conclusion

We believe this design is a generic approach that can be used to improve the user experience during service/platform composition, shortening service discovery time by 50x. We have shown that the UWB layer-2 discovery mechanism effectively supports the composition framework described in [10], enhancing a user's capability to build computer systems on the fly — wirelessly connecting distributed components using service abstractions. In conclusion, we observe that layer-2 service discovery, delivering service descriptions in less than 200 ms (greater than any measured in Fig. 7b), will be perceived by users as nearly instantaneous, greatly facilitating the ease of use for the overall system.

References

[1] B. O'Hara and A. Petrick, *IEEE 802.11 Handbook: A Designer's Companion*, IEEE Press, 2005.
[2] J. C. Haartsen, "The Bluetooth Radio System," *IEEE Pers. Commun.*, vol. 7, no. 1, Feb. 2000, pp. 28–36.
[3] G. R. Hiertz, "High Rate Ultra Wideband PHY and MAC Standard Multiband OFDM Alliance — The Next Generation of Wireless Personal Area Networks," *AWWC, 2005 IEEE Sarnoff/Symp.*, Apr. 18–19, 2005, pp. 208–14.
[4] R. Want *et al.*, "Personal Servers: Changing the Way We Think about Ubiquitous Computing," *Proc. Ubicomp 2002, 4th Int'l. Conf.*, Springer LNCS 2498, Goteborg, Sweden, Sept. 30–Oct. 2, 2002, pp. 194–209.
[5] A. Fox *et al.*, "Integrating Informatio Appliances into an Interactive Workspace," *IEEE Comp. Graphics and Apps.*, vol. 20, no. 3, May 2000, pp. 54–65.
[6] K. W. Edwards, "Discovery Systems in Ubiquitous Computing," *IEEE Pervasive Computing*, vol. 5, no. 2, Apr.–June 2006, pp. 70–77.
[7] M. Jeronimo and J. Weast, *UPnP Design by Example: A Software Developer's Guide to Universal Plug and Play*, Intel Press, 2003.
[8] S. Cheshire and D. H. Steinburg, *Zero Configuration Networking: The Definitive Guide*, O'Reilly Media, 2006.
[9] M. Moore, "Designing for Ultra Low-Power: Ultra Wideband (UWB) in Handheld Devices," http://www.portabledesign.com/article?article_id=88, Nov. 2007.
[10] R. Want *et al.*, "Dynamic Composable Computing," *ACM HotMobile '08*, Napa, CA, Feb. 25–26, 2008.

Biographies

Shivani Sud (shivani.a.sud@intel.com) is a senior architect with background in systems software and networking protocols. She is working with the Ubiquity Group in Intel Research. Her prior research work was around platform-based Identity management investigating platform security options for existing and in-research Intel client architectures. Before that she has worked in other research and product groups working on software architecture and design and development. Her areas of technical specialization include Intel architectures, Ultra Wide Band, personal computing, IP networking protocol stacks and distributed computing. She received her M.S. in Computer Science and BS in Physics from University of Delhi, India.

Roy Want [F] (roy.want@intel.com) is a Senior Principal Engineer at Intel Research, Santa Clara, California, and leader of the Ubiquity group. His research interests include mobile & ubiquitous computing, wireless protocols, hardware design, embedded systems, distributed systems, automatic identification and micro-electromechanical systems (MEMS). He received his BA in computer science from Churchill College, Cambridge University, UK in 1983 and continued research at Cambridge into reliable distributed multimedia-systems. While at Olivetti Research (1988–1991) he developed the first in-building location system called the Active Badge, launching his interest in location-based services. He joined Xerox PARC's Ubiquitous Computing program in 1991 and led a project called PARCTab, one of the first context-aware computer systems. At PARC he managed the Embedded Systems area and earned the title of Principal Scientist. He joined Intel Research in 2000 as a Principal Engineer. He is also the author, or co-author, of more than 60 publications in the field of mobile and distributed systems; and has over 55 patents issued in these areas. He is very involved in the research community through program committees and invited talks. He is Editor-in-chief for IEEE Pervasive Computing and a Fellow of the ACM.

Trevor Pering (trevor.pering@intel.com) is a senior researcher with the Ubiquity Group in Intel Research. His research has focused on many aspects of mobile and ubiquitous computing, including hardware design, wireless systems, and mobile system use models. His background is in Computer Science and Electrical Engineering, having received his B.S. and Ph.D. from the University of California, Berkeley - but he has a strong inclination towards developing a compelling user experience supported by emerging technology. His initial research at Berkeley focused on the InfoPad wireless tablet and Dynamic Voltage Scaling for microprocessor. At Intel, he has been following this line of mobile computing research, starting with the Persona project, several years developing the Personal Server concept and Stargate hardware platform, and is currently working on Dynamically Composable Computing. Outside of work, he enjoys music, the outdoors, and exploring funky restaurants and cafes.

Barbara Rosario (barbara.rosario@intel.com) has a master degree in physics from University of Trieste, Italy. She then worked as a researcher at the University of Cambridge UK and the Media Lab, MIT. She then moved to Berkeley, CA where she received a Ph.D. from the Information School, UC Berkeley. She now works at Intel Research, focusing on speech interfaces, natural language processing and machine learning.

Kent Lyons (kent.lyons@intel.com) is a Research Scientist at Intel Research. He received his Ph.D. in computer science in 2005 from the Georgia Institute of Technology. His research interests include Human-Computer Interaction, Mobile and Ubiquitous Computing, and Wearable Computing. A technologist at heart, his research focuses on applying novel technologies in a user centered way; thereby bridging the gap between these two research styles. His work has explored the collocated use of mobile devices as well as the I/O challenges associated with different kinds of mobile computers. More generally, he is interested in exploring the integration of mobile devices into everyday life through the use of HCI. He has authored numerous HCI papers on mobile and wearable computing and presented user interface work at CHI, UIST, ISWC and Mobile HCI. He has also had extensive experience with the everyday use of wearable computers having worn one himself for many years.