



Overview of UPnP* AV Architecture

A Digital Media Distribution Technology for the Home

**Research &
Development
at Intel**

(Version 1.00, 07-02-2003)

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* UPnP is a certification mark of the UPnP Implementers Corp. Other names and brands may be claimed as the property of others.

Copyright © 2003 Intel Corporation

This publication was downloaded from: <http://developer.intel.com/>

Overview of UPnP AV Architecture

July 2003

Abstract

The proliferation of digital technologies within the home is paving the way for significant enhancements to the user's entertainment experience. A fundamental aspect of these new experiences is the ability to enjoy rich multimedia content (e.g., videos, music, pictures) in any location throughout the home, regardless of where the content is physically stored. Today's home networking technologies provide a solid foundation for distributing content throughout the home and achieving this type of "anytime, anywhere" access to content.

In reality, however, most media distribution networking technologies are not suitable for mass-market adoption due to the complexity associated with their installation, configuration, and management. This paper describes a media distribution technology called UPnP* AV (Audio/Visual) architecture. UPnP AV architecture provides a foundation for building consumer-oriented AV products that provide consumers with a simple out-of-the-box installation and configuration experience.

Introduction

The increasing availability of numerous types of digital devices allows consumers to access a wide range of rich multimedia content from a variety of sources. These sources include Personal Computers (PCs), the Internet, digital terrestrial broadcast receivers, satellite receivers, and Consumer Electronic (CE) devices such as CD/DVD players, camcorders, digital still cameras, and portable audio players. The physical location of these content sources is often not the user's preferred location for experiencing them. For example, users may prefer to listen to their MP3 music collection on their living room stereo instead of having to sit in the den where

the PC is typically located.

Although today's home networking technologies provide a solid foundation for distributing digital entertainment content throughout the home, some key barriers remain that prevent the wide-spread adoption of media distribution solutions by consumers. Most media distribution systems are custom, end-to-end solutions based on proprietary technologies that require a trained specialist to set up and configure. These factors prevent such solutions from reaching price-points and convenience levels that are essential for mass-market adoption of these systems.

The key factor affecting both price-point and convenience is the need for a professional installer. In order to enable wide-spread adoption, these systems need to be installable and configurable by the end-user, without help from a service representative—just like today's VCR. This type of "out-of-box" installation experience requires individual devices to be self-configurable. Additionally, devices need to discover other devices on the network that they can interact with in an autonomous manner.

This type of self-configuring system requires, among other things, the adoption of a single interoperability technology that is pervasive throughout the industry (i.e., adopted by a large number of device manufacturers and supported by a wide range of devices). There have been many efforts in the past to define such interoperability technologies. Although these efforts adequately solved many of the technical issues, their lack of critical mass within the industry limits their success in the marketplace.

A survey of the current status of various interoperability technologies reveals that the UPnP specification offers the greatest promise for an

Overview of UPnP AV Architecture

July 2003

interoperability technology that will actually be adopted and deployed by the key market-making leaders within the industry. The following sections give an overview of UPnP technology and describe how UPnP Audio Visual (AV) architecture provides media distribution services within the home that allow mass-market consumers to realize a variety of compelling end-user scenarios.

The UPnP Technology

UPnP technology runs on top of an IP/TCP (Internet Protocol/Transmission Control Protocol) networking layer. The UPnP device architecture specification [1] defines general interoperability mechanisms that enable self-configuring devices to create an ad-hoc, self-discovering system of interoperable network devices. This specification defines mechanisms for automatic address configuration, device discovery, command/control, and eventing. The UPnP specification also defines “Presentation Pages” that allow devices to expose dedicated Web pages for user-initiated interaction with a specific device.

UPnP Architecture Fundamentals

UPnP technology uses existing Internet standards including TCP/IP, HTTP, SSDP, SOAP, GENA, and XML. These open specifications provide the communication infrastructure of UPnP architecture. Although UPnP architecture consists of a peer-to-peer network, nodes on the network communicate with each other in a client-server manner. Clients are called *Control Points* (CP) and typically provide a User Interface (UI) for end-users. Servers are called *Controlled Devices* (henceforth, called devices) and by definition expose a well-defined set of *services*, each of which corresponds to a functional component of the

device. Each service defines a set of *state variables* and *actions* that allow Control Points to obtain the current state of the device and to control the device’s operation. Invoking an action usually causes a change in the internal state of the device that affects the value of certain state variables. In all cases, Control Points invoke actions, and devices respond to actions that are received.

In order to enable autonomous device interoperability, members of the UPnP Forum [2] constructed a set of device and service definitions (a.k.a., templates) that are be used to model various common devices. Since the behavior of these device and service templates is well defined, Control Points can interoperate with any device that implements one of the defined device templates. In this manner, Control Points and devices can be built independently by different manufacturers with the assurance that they will interoperate according to the functionality defined by the corresponding UPnP device/service templates.

Network Addressing

Since UPnP architecture is built on top of the Internet Protocol (IP), each node in the network requires a unique IP address. This address is assigned either via a Dynamic Host Configuration Protocol (DHCP) Server, or via the “Auto-IP” protocol if a DHCP Server is not available. When a DHCP service becomes available, all nodes are required to obtain an address from it. Once a device or a Control Point has been assigned an address, it is considered “added” to the network.

Discovery

When a Control Point is added to the network, it needs to discover (i.e., locate) the devices in the network that it is capable of controlling. This is

Overview of UPnP AV Architecture

July 2003

accomplished via the Simple Device Discovery Protocol (SDDP) by broadcasting a discovery request that identifies the functional capabilities (i.e., the device types) that the Control Point wants to control. Any device that exposes those capabilities responds to the request by identifying itself to the Control Point.

The device response contains the URL of the “XML device description document,” which identifies the services that the device implements, as well as the specific actions and state variables that are supported by each service. By parsing this information, the Control Point is able to determine the exact capabilities of each device. This allows a Control Point to determine if it wants to interact with and control a particular device.

When a new device is added to the network, the device broadcasts an identification notification to the network. This notification informs existing Control Points that a new device has been added to the network and is available to be controlled. The notification information includes the URL of the new device’s description document, as described above.

Command/Control

Once a Control Point has determined that it wants to control a particular device, the Control Point uses the Simple Object Access Protocol (SOAP) to invoke the actions that are exposed by the device’s services. The behavior of each action is well defined by the service template document.

Eventing

As the internal state of a device changes, either in response to an action or via some internal condition, the device can inform one or more Control Points of

the state change using Generic Event Notification Architecture (GENA). With this protocol, Control Points that need to be informed of state changes within a particular device must register with that device to receive event notifications. A given device may be monitored by multiple Control Points. When an internal state change occurs, the device sends an event notification to each Control Point that has registered with the device. This event notification includes an identification of the state variable that has changed, along with its new value. The set of state variables that are evented by the device is defined in each of the service templates that are supported by the device. Additionally, each evented state variable may be moderated such that rapid changes in that state variable do not cause excessive network traffic.

The UPnP AV Specifications

The UPnP Audio Visual (AV) specifications [3] define a set of UPnP device and service templates that specifically target Consumer Electronic (CE) devices such as TVs, VCRs, DVD players, stereo systems, MP3 players, and PCs. In this context, a CE device refers to any device that interacts with entertainment content (e.g., movies, audio, and still images).

In today’s non-networked CE environment, CE devices interoperate with each other using dedicated cables (e.g., RCA cables). The UPnP AV specification enables CE devices to use a digital network instead of dedicated analog cables to interoperate with each other. This network-wide interoperability allows CE devices to distribute entertainment content throughout the Digital Home network.

UPnP AV Fundamentals

Overview of UPnP AV Architecture

July 2003

The UPnP AV architecture shown in Figure 1 defines the three main logical entities that constitute UPnP AV architecture: a Media Server, a Media Renderer, and a Control Point. The Media Server has access to entertainment content and can send that content to other UPnP AV devices via the network. A Media Renderer is a UPnP AV device that is able to receive external content from the network and render it on its local hardware. An AV Control Point coordinates the operation of the Media Server and Media Renderer in order to accomplish the desires of the end user.

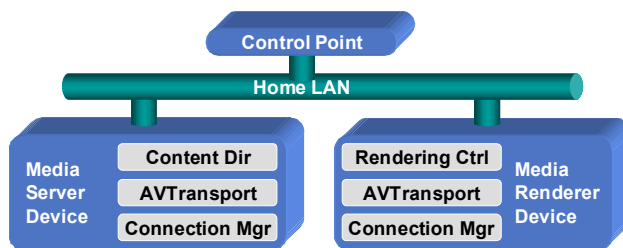


Figure 1: UPnP AV Architecture

As described later in this section, Media Servers and Media Renderers implement a set of UPnP AV services. These services provide command and control functions that allow a Control Point to set up and configure the Media Server and Media Renderer for transferring the desired content from the Media Server to the Media Renderer.

The Control Point is only involved in command and control operations. It is not involved in the actual transfer of the content. Therefore, the Control Point, and hence the entire AV architecture, is not dependent on any particular transfer protocol and/or content format.

Since the AV architecture can accommodate various transfer protocols and content formats, Media Servers and Media Renderers can transfer the desired content using any transfer protocol and data

format that they *both* support. As part of its set-up and configuration responsibilities, the Control Point must identify and select the protocol and format to be used. However, since the Control Point is not involved in the actual transfer, it does not need to implement the transfer protocol or support the content formats.

Although the AV architecture defines three logical entities—Media Server, Media Renderer, and Control Point—a physical device may contain any combination of them. For example, many renderers are likely to include an embedded Control Point so that the user can control the operation using the same device that is rendering the content.

UPnP AV Control Points

As described above, AV Control Points control the operation of the Media Servers and Media Renderers so that the user can render the specific content on a particular rendering device. In most end-user scenarios, the Control Point uses a variation of the following algorithm:

- Locate the existing Media Server and Media Renderer devices in the network, i.e., discovery.
- Enumerate the available content for the user to choose from, i.e., content enumeration.
- Query the server and renderer to find a common transfer protocol and content format for the selected content, i.e., protocol/format negotiation.
- Configure the server and renderer with the desired content and selected protocol/format, i.e., server/renderer setup.
- Initiate the transfer of the content according to the desires of the users, such as play, pause, and seek, i.e., control content flow.
- Adjust the rendering characteristics such as volume, brightness, and so forth, i.e., control

Overview of UPnP AV Architecture

July 2003

rendering characteristics.

The Control Point accomplishes this general algorithm by invoking various actions on UPnP AV services exposed by the server and renderer. In this manner, the Control Point performs content distribution tasks desired by the user.

Media Server

A Media Server is a device that has access to entertainment content (e.g., videos, music, pictures) and can send that content to another device for rendering. Media Servers include familiar devices such as VCRs, set-top boxes (including cable, satellite, and digital broadcast receivers), camcorders, CD/DVD players/jukeboxes, radio tuners, TV tuners, still-image cameras, etc.

Media Servers expose a set of UPnP services including the Content Directory, Connection Manager, and (optionally) the AV Transport services. The Content Directory service allows a Control Point to discover and enumerate all of the content that is accessible by the server. The Connection Manager service allows the Control Point to negotiate and select the transfer protocol and data format that will be used by the server and renderer to transfer the desired content. The (optional) AV Transport service allows control of the flow of the content (e.g., play, stop, and pause) when using a “push” oriented transfer protocol (e.g., IEEE-1394).

Media Renderer

A Media Renderer is a device that can receive content from another device (e.g., a Media Server) and render it using local hardware. This includes familiar devices such as a TV, a stereo system, a set of speakers, an Electronic Picture Frame (EPF), etc. Innovative renderers can use any type of output hardware that can be controlled by the incoming

content. For example, a “Music Fountain” can generate dancing streams of water based on the content of a song (e.g., the song’s varying loudness and/or audio frequencies).

Each Media Renderer exposes a set of UPnP services including the Rendering Control, Connection Manager, and (optionally) the AV Transport (AVT) services. The Rendering Control service controls how the content is rendered (e.g., volume, brightness, etc.). As with the Media Server, the Connection Manager service is used to negotiate a common protocol/format. The (optional) AV Transport service is used to control the flow of the content when using a “pull” oriented transfer protocol (e.g., HTTP-GET).

Content Directory Service

The Content Directory Service (CDS) allows Control Points to discover and enumerate content that is accessible by a Media Server. CDS “content” objects include individual “content items,” which represent individual pieces of content (such as a song, a video clip, or a photo), and “content containers,” which represent collections of items (such as a playlist, a CD, a DVD, or a photo album). Each CDS object, either an item or container, includes meta-data that describes various attributes of the object (such as title, artist, duration, etc.).

CDS provides both *Browse* and *Search* capabilities. Control Points that browse a CDS begin at the root of the CDS hierarchy and iteratively examine the structure, container by container, until the desired content item is found. This is similar to how a file system is used to locate a file that is nested several layers down from the root directory. Control Points typically use this mechanism when the user does not immediately have a particular content item in mind.

Alternatively, a Control Point can use the search

Overview of UPnP AV Architecture

July 2003

capability of the CDS to locate all of the items/containers that possess certain attributes (i.e., certain meta-data values such as “creator=Disney”).

Part of the meta-data for each object is a list of transfer protocol and data format combinations that are supported for that piece of content. This information is used by the Control Point in conjunction with the Connection Manager service on the target Media Renderer to determine which protocols and formats can be used to transfer the content to the renderer. Each protocol/format combination is identified by a unique Universal Resource Identifier (URI) that is used by the server and renderer to uniquely identify the content, protocol, and format that is to be used during the transfer.

The data structures defined by CDS dictate the over-the-wire representation of content items/containers and associated meta-data. CDS does not define the Media Server’s internal storage mechanisms or structures. Media Servers can store CDS information using any appropriate mechanism. For example, a server may use a full-featured database system to store its CDS content hierarchy and to provide a rich set of meta-data for each object. Other servers may maintain their CDS information using only a directory/file hierarchy of their internal file system. In this case, the breadth of the meta-data for each object will be limited to the information that is stored by the file system for each directory/file.

As CDS requests are made by the Control Point, the server converts those requests into a set of operations carried out by the underlying database used to store the CDS hierarchy and meta-data. Depending on the server’s implementation, this may be a set of relational database operations or a set of system calls to the local file system.

Rendering Control Service

The Rendering Control Service (RCS) is implemented on a Media Renderer in order to provide the Control Point with a mechanism to control how the content is rendered (e.g., volume, brightness, contrast, etc.). These functions are directly related to the capabilities of the output hardware on the renderer.

The internal logic of RCS is fairly simple. As RCS actions are invoked, the RCS converts the requested adjustment to the corresponding hardware request as needed.

Connection Manager Service

The Connection Manager (CM) service is implemented on both the server and renderer. The primary purpose of the CM service is to allow the Control Point to identify and select the common protocol/format that will be used to transfer the desired content from the server to the renderer.

The actions defined by the CM service provide a standardized interface to obtain the capabilities of the network and media codec subsystems of the server/renderer. An implementation of CM must be able to enumerate and configure the transfer protocols that are supported by the device’s network subsystem and the data formats that are supported by the device’s media codecs.

In order to enumerate the list of supported protocols and formats, the CM service on most fixed-function devices, such as traditional CE equipment, is fairly simple since the set of supported protocols and formats is fixed (i.e., it is known what media formats the device is designed to support). For other general purpose devices, such as the PC, the CM service exposes the network protocols and codec models that have been dynamically installed on the device.

Overview of UPnP AV Architecture

July 2003

When preparing to transfer a piece of content from the server to the renderer, the CM service must be able to setup and configure its network and codec subsystems according to the requested mechanism. On a server device, this may require constructing a streaming data path from the device's storage subsystem, through one of its codec subsystems, and out through its network subsystem. On a renderer device, this may involve constructing a streaming data path from the device's network subsystem, through the appropriate codec, to the device's output hardware. In many implementations, the CM service uses a set of pluggable modules to construct this data path for the content stream. A popular example of a pluggable media streaming engine is DirectX* from Microsoft*. With this technology, individual filters correspond to particular media streaming functions (e.g., capture a data stream from the network, decode it, etc.). Individual filters are plugged together to form a complete data path called a filter graph.

AV Transport Service

The AV Transport (AVT) service provides a number of actions that allow a Control Point to control the flow of the content. This includes many of the familiar operations typically associated with the mechanical "tape transport" mechanism implemented on most VCRs. This includes end-user operations such as Play, Stop, Pause, Seek (e.g., FF/REW), etc.

AVT also provides the mechanism which the Control Point uses to identify the content that is to be played. This is done by passing the unique URI obtained from the Content Directory Service for the desired content and the selected protocol and format.

Depending on which transfer protocol is used to transfer the content, either the server or the renderer will provide an instance of the AVT service. If the selected protocol is a "pull" model (e.g., HTTP GET) then the renderer provides an instance of AVT to control the flow of the content (e.g., play, pause, and seek). If the selected protocol is a "push" model (e.g., IEEE-1394) then the server provides an instance of AVT.

The internal implementation of AVT hooks into the device's media streaming subsystem in order to configure it to access and stream the desired content and to control that content flow as directed by the Control Point (i.e., the end-user). In most cases, the internal logic of the AVT service is fairly straightforward. When an AVT action is invoked, AVT invokes the corresponding operation(s) on the internal media streaming subsystem. When the device is using a pluggable media streaming technology like DirectX*, AVT simply invokes the appropriate method(s) on the filter graph that is associated with the content.

UPnP AV Control Point Algorithm

As described above, UPnP AV architecture defines the external interfaces of the Media Server and Media Renderer so that a Control Point can manage the distribution of content as desired by the end-user. However, the AV architecture does not define any of the internal structures of the server, renderer or Control Point. This is left entirely to the implementer. Nevertheless, in practice there are some general implementation models that will be commonplace. We outline some of these models below.

When a Control Point joins the network, it locates all of the media servers and media renderers in the network. It does this using Simple Service

Overview of UPnP AV Architecture

July 2003

Discovery Protocol (SSDP). In order to locate Media Servers in the network, the Control Point issues an SSDP IP-multicast request packet to locate any UPnP device that implements the UPnP AV Media Server device template. All devices that implement the Media Server template must respond to the request with the URL of their description document. Media Renderers are located in a similar manner except using the Media Renderer device template.

After the servers and renderers are located, the Control Point obtains and parses each device's XML description document to determine the device's exact capabilities (i.e., its UPnP services, actions, and state variables). If the device implements the desired capabilities, the Control Point continues to interact with it as described below.

At some point after the Control Point initializes itself, it may display an initial User Interface (UI) so that the end user can interact with the Control Point. The contents and layout of the UI are vendor-dependent which allows for a wide-variety of innovation and product differentiation.

For each Media Server that is found, the Control Point uses the server's Content Directory Service (CDS) to enumerate the content that is available from that server. Control Points often collect CDS information from multiple servers and aggregate it into a single "whole home" view of all of the content that is available from within the home, regardless of which Media Server can provide it.

Depending on the Control Point's UI, the Control Point will either browse through the CDS information, perform searches on it, or a combination. Once the Control Point has received and processed the returned data, the Control Point

updates its UI.

After the user has selected the desired content, the Control Point determines the transfer protocols and data formats supported for that particular piece of content. This is done by examining the CDS meta-data for the selected item. Using the Connection Manager service on each renderer, the Control Point can obtain the set of protocols/formats that are supported by those devices. The Control Point then compares the protocol/format information from the server's CDS and the renderer's CM to determine which renderer(s) is capable of rendering the desired content.

If a common protocol/format has been identified, the Control Point invokes the CM services on both the server and renderer to inform each device of the target protocol/format. In response, the CM sets up and configures its internal network and media streaming subsystems based on the common protocol/format that has been chosen.

As a result of configuring each device, either the server or renderer returns an instance of the AVT service that is associated with the data path that has just been set up. The Control Point uses the returned AVT to specify the content item that is to be transferred from the server to the renderer.

When the user indicates the desired operation to be performed on the content (e.g., play) the appropriate actions on the AVT service are invoked. After the content begins to play, the user may select other operations, such as Stop, Pause, etc.

As the content is being rendered, the Control Point may provide a set of UI components that allow the user to control how the content is rendered. This includes various rendering characteristics such as volume and brightness. As the user adjusts various rendering characteristics, the Control Point invokes

Overview of UPnP AV Architecture

July 2003

the appropriate action on the Rendering Control Service (RCS).

Conclusion

UPnP AV Architecture defines the foundational interfaces that enable device manufacturers to develop and deploy multimedia products that interoperate with devices from other manufactures. Since AV Architecture already defined the fundamental interoperability mechanisms, manufacturers are free to focus their energies on building innovative capabilities into their products. This helps simplify the development process and allow vendors to provide self-configuring, interoperable products to the marketplace with lower development costs. The combination of lower development and installations costs enables products to be brought to the marketplace at price-points and convenience levels that allow for mass-market adoption of these products. Consequently, UPnP AV Architecture makes it possible for mass-market consumers to finally realize the compelling benefits of being able to access rich multimedia content “anytime, anywhere”.

References

- [1] UPnP Device Architecture, Microsoft Corporation*,
http://www.upnp.org/download/UPnPDA10_20000613.htm
- [2] UPnP Forum, <http://www.upnp.org>
- [3] UPnP AV 1.0 Specifications, UPnP Forum,
<http://www.upnp.org/standardizeddcps/mediaserver.asp>
- [4] UPnP Technology, Intel Research and Development,
<http://www.intel.com/labs/connectivity/upnp/>