ISO/IEC 29341-1

Edition 2.0 2011-09

INTERNATIONAL STANDARD

colour inside

Information technology – UPnP device architecture – Part 1: UPnP Device Architecture Version 1.0

ISO/IEC 29341-1:2011(E)

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub

The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

ISO/IEC 29341-1

Edition 2.0 2011-09

INTERNATIONAL STANDARD

colour inside

Information technology – UPnP device architecture – Part 1: UPnP Device Architecture Version 1.0

INTERNATIONAL ELECTROTECHNICAL COMMISSION

PRICE CODE V

ICS 35.200

ISBN 978-2-88912-676-7

CONTENTS

INFORMATION TECHNOLOGY – UPNP DEVICE ARCHITECTURE –

Part 1: UPnP Device Architecture Version 1.0

FOREWORD

1) ISO (International Organization for Standardization) and IEC (International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards. Their preparation is entrusted to technical committees; any ISO and IEC member body interested in the subject dealt with may participate in this preparatory work. International governmental and non-governmental organizations liaising with ISO and IEC also participate in this preparation.

2) In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

3) The formal decisions or agreements of IEC and ISO on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC and ISO member bodies.

4) IEC, ISO and ISO/IEC publications have the form of recommendations for international use and are accepted by IEC and ISO member bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC, ISO and ISO/IEC publications is accurate, IEC or ISO cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.

5) In order to promote international uniformity, IEC and ISO member bodies undertake to apply IEC, ISO and ISO/IEC publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any ISO/IEC publication and the corresponding national or regional publication should be clearly indicated in the latter.

6) ISO and IEC provide no marking procedure to indicate their approval and cannot be rendered responsible for any equipment declared to be in conformity with an ISO/IEC publication.

7) All users should ensure that they have the latest edition of this publication.

8) No liability shall attach to IEC or ISO or its directors, employees, servants or agents including individual experts and members of their technical committees and IEC or ISO member bodies for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication of, use of, or reliance upon, this ISO/IEC publication or any other IEC, ISO or ISO/IEC publications.

9) Attention is drawn to the normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

10) Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 29341-1 was prepared by UPnP Forum Steering committee[1], was adopted, under the fast track procedure, by subcommittee 25: Interconnection of information technology equipment, of ISO/IEC joint technical committee 1: Information technology.

This International Standard replaces ISO/IEC 29341-1, first edition, published in 2008, and constitutes a technical revision.

The list of all currently available parts of the ISO/IEC 29341 series, under the general title *Information technology – UPnP device architecture*, can be found on the IEC web site.

This International Standard has been approved by vote of the member bodies, and the voting results may be obtained from the address given on the second title page.

---

1 UPnP Forum Steering committee, UPnP Forum, 3855 SW 153rd Drive, Beaverton, Oregon 97006 USA. See also "Introduction".

Introduction

What is UPnP™[1] Technology?

UPnP™ technology defines an architecture for pervasive peer-to-peer network connectivity of intelligent appliances, wireless devices, and PCs of all form factors. It is designed to bring easy-to-use, flexible, standards-based connectivity to ad-hoc or unmanaged networks whether in the home, in a small business, public spaces, or attached to the Internet. UPnP technology provides a distributed, open networking architecture that leverages TCP/IP and the Web technologies to enable seamless proximity networking in addition to control and data transfer among networked devices.

The UPnP Device Architecture (UDA) is more than just a simple extension of the plug and play peripheral model. It is designed to support zero-configuration, "invisible" networking, and automatic discovery for a breadth of device categories from a wide range of vendors. This means a device can dynamically join a network, obtain an IP address, convey its capabilities, and learn about the presence and capabilities of other devices. Finally, a device can leave a network smoothly and automatically without leaving any unwanted state behind.

The technologies leveraged in the UPnP architecture include Internet protocols such as IP, TCP, UDP, HTTP, and XML. Like the Internet, contracts are based on wire protocols that are declarative, expressed in XML, and communicated via HTTP. Using internet protocols is a strong choice for UDA because of its proven ability to span different physical media, to enable real world multiple-vendor interoperation, and to achieve synergy with the Internet and many home and office intranets. The UPnP architecture has been explicitly designed to accommodate these environments. Further, via bridging, UDA accommodates media running non-IP protocols when cost, technology, or legacy prevents the media or devices attached to it from running IP.

What is "universal" about UPnP technology? No device drivers; common protocols are used instead. UPnP networking is media independent. UPnP devices can be implemented using any programming language, and on any operating system. The UPnP architecture does not specify or constrain the design of an API for applications. OS vendors may create APIs that suit their customers' needs.

UPnP™ Forum

The UPnP Forum is an industry initiative designed to enable easy and robust connectivity among stand-alone devices and PCs from many different vendors. The UPnP Forum seeks to develop standards for describing device protocols and XML-based device schemas for the purpose of enabling device-to-device interoperability in a scalable networked environment.

The UPnP Implementers Corporation (UIC) is comprised of UPnP Forum member companies across many industries who promote the adoption of uniform technical device interconnectivity standards and testing and certifying of these devices. The UIC develops and administers the testing and certification process, administers the UPnP logo program, and provides information to UIC members and other interested parties regarding the certification of UPnP devices. The UPnP device certification process is open to any vendor who is a member of the UPnP Forum and UIC, has paid the UIC dues, and has devices that support UPnP functionality. For more information, see http://www.upnp-ic.org.

The UPnP Forum has set up working committees in specific areas of domain expertise. These working committees are charged with creating proposed device standards, building sample implementations, and building appropriate test suites. This document indicates specific technical decisions that are the purview of UPnP Forum working committees.

UPnP vendors can build compliant devices with confidence of interoperability and benefits of shared intellectual property and the logo program. Separate from the logo program, vendors may also build devices that adhere to the UPnP Device Architecture defined herein without a

[1] UPnP™ is a certification mark of the UPnP™ Implementers Corporation.

formal standards procedure. If vendors build non-standard devices, they determine technical decisions that would otherwise be determined by a UPnP Forum working committee.

In this document

The UPnP Device Architecture (formerly known as the DCP Framework) contained herein defines the protocols for communication between controllers, or *control points*, and devices. For discovery, description, control, eventing, and presentation, the UPnP Device Architecture uses the following protocol stack (the indicated colors and type styles are used throughout this document to indicate where each protocol element is defined):

<table>
<tr><td colspan="4">UPnP vendor [purple-italic]</td></tr>
<tr><td colspan="4">UPnP Forum [red-italic]</td></tr>
<tr><td colspan="4">UPnP Device Architecture [green-bold]</td></tr>
<tr><td colspan="2">SSDP [blue]</td><td>SOAP [blue]</td><td>GENA [navy-bold]</td></tr>
<tr><td>HTTPMU (multicast) [black]</td><td>HTTPU (unicast) [black]</td><td>HTTP [black]</td><td>HTTP [black]</td></tr>
<tr><td colspan="2">UDP [black]</td><td colspan="2">TCP [black]</td></tr>
<tr><td colspan="4">IP [black]</td></tr>
</table>

At the highest layer, messages logically contain only UPnP vendor-specific information about their devices. Moving down the stack, vendor content is supplemented by information defined by UPnP Forum working committees. Messages from the layers above are hosted in UPnP-specific protocols such as the Simple Service Discovery Protocol (SSDP) and the General Event Notification Architecture (GENA) defined in this document, and others that are referenced. The above messages are delivered via HTTP, either a multicast or unicast variety running over UDP, or the standard HTTP running over TCP. Ultimately, all messages above are delivered over IP. The remaining clauses of this document describe the content and format for each of these protocol layers in detail. For reference, colors in [square brackets] above indicate which protocol defines specific message components throughout this document.

Two general classifications of devices are defined by the UPnP architecture: controlled devices (or simply “devices”), and control points. A controlled device functions in the role of a server, responding to requests from control points. Both control points and controlled devices can be implemented on a variety of platforms including personal computers and embedded systems. Multiple devices, control points, or both may be operational on the same network endpoint simultaneously.

The foundation for UPnP networking is IP addressing. Each device must have a Dynamic Host Configuration Protocol (DHCP) client and search for a DHCP server when the device is first connected to the network. If a DHCP server is available, i.e., the network is managed, the device must use the IP address assigned to it. If no DHCP server is available, i.e., the network is unmanaged, the device must use Auto IP to get an address. In brief, Auto IP defines how a device intelligently chooses an IP address from a set of reserved addresses and is able to move easily between managed and unmanaged networks. If during the DHCP transaction, the device obtains a domain name, e.g., through a DNS server or via DNS forwarding, the device should use that name in subsequent network operations; otherwise, the device should use its IP address.

Given an IP address, Step 1 in UPnP networking is *discovery*. When a device is added to the network, the UPnP discovery protocol allows that device to advertise its services to control points on the network. Similarly, when a control point is added to the network, the UPnP discovery protocol allows that control point to search for devices of interest on the network. The fundamental exchange in both cases is a discovery message containing a few, essential specifics about the device or one of its services, e.g., its type, identifier, and a pointer to more detailed information. The clause on Discovery below explains how devices advertise, how control points search, and details of the format of discovery messages.

Step 2 in UPnP networking is *description*. After a control point has discovered a device, the control point still knows very little about the device. For the control point to learn more about the device and its capabilities, or to interact with the device, the control point must retrieve the device's description from the URL provided by the device in the discovery message. Devices may contain other, logical devices, as well as functional units, or *services*. The UPnP description for a device is expressed in XML and includes vendor-specific, manufacturer information like the model name and number, serial number, manufacturer name, URLs to vendor-specific Web sites, etc. The description also includes a list of any embedded devices or services, as well as URLs for control, eventing, and presentation. For each service, the description includes a list of the commands, or *actions*, the service responds to, and parameters, or *arguments*, for each action; the description for a service also includes a list of variables; these variables model the state of the service at run time, and are described in terms of their data type, range, and event characteristics. The clause on Description below explains how devices are described and how those descriptions are retrieved by control points.

Step 3 in UPnP networking is *control*. After a control point has retrieved a description of the device, the control point can send actions to a device's service. To do this, a control point sends a suitable control message to the control URL for the service (provided in the device description). Control messages are also expressed in XML using the Simple Object Access Protocol (SOAP). Like function calls, in response to the control message, the service returns any action-specific values. The effects of the action, if any, are modeled by changes in the variables that describe the run-time state of the service. The clause on Control below explains the description of actions, state variables, and the format of control messages.

Step 4 in UPnP networking is *eventing*. A UPnP description for a service includes a list of actions the service responds to and a list of variables that model the state of the service at run time. The service publishes updates when these variables change, and a control point may subscribe to receive this information. The service publishes updates by sending event messages. Event messages contain the names of one of more state variables and the current value of those variables. These messages are also expressed in XML. A special initial event message is sent when a control point first subscribes; this event message contains the names and values for all evented variables and allows the subscriber to initialize its model of the state of the service. To support scenarios with multiple control points, eventing is designed to keep all control points equally informed about the effects of any action. Therefore, all subscribers are sent all event messages, subscribers receive event messages for all evented variables that have changed, and event messages are sent no matter why the state variable changed (either in response to a requested action or because the state the service is modeling changed). The clause on Eventing below explains subscription and the format of event messages.

Step 5 in UPnP networking is *presentation*. If a device has a URL for presentation, then the control point can retrieve a page from this URL, load the page into a browser, and depending on the capabilities of the page, allow a user to control the device and/or view device status. The degree to which each of these can be accomplished depends on the specific capabilities of the presentation page and device. The clause on Presentation below explains the protocol for retrieving a presentation page.

Audience

The audience for this document includes UPnP device vendors, members of UPnP Forum working committees, and anyone else who has a need to understanding the technical details of UPnP protocols.

This document assumes the reader is familiar with the HTTP, TCP, UDP, IP family of protocols; this document makes no attempt to explain them. This document also assumes most readers will be new to XML, and while it is not an XML tutorial, XML-related issues are addressed in detail given the centrality of XML to the UPnP device architecture. This document makes no assumptions about the reader's understanding of various programming or scripting languages.

Required vs. recommended

In this document, features are described as Required, Recommended, or Optional as follows:

Required (or Must or Shall).

These basic features must be implemented to comply with the UPnP Device Architecture. The phrases “must not” and “shall not” indicate behavior that is prohibited that if performed means the implementation is not in compliance.

Recommended (or Should).

These features add functionality supported by the UPnP Device Architecture and should be implemented. Recommended features take advantage of the capabilities of the UPnP Device Architecture, usually without imposing major cost increases. Notice that for compliance testing, if a recommended feature is implemented, it must meet the specified requirements to be in compliance with these guidelines. Some recommended features could become requirements in the future. The phrase “should not” indicates behavior that is permitted but not recommended.

Optional (or May).

These features are neither required nor recommended by the UPnP Device Architecture, but if the feature is implemented, it must meet the specified requirements to be in compliance with these guidelines. These features are not likely to become requirements in the future.

Acronyms

Acronym	Meaning	Acronym	Meaning
ARP	Address Resolution Protocol	SOAP	Simple Object Access Protocol
CP	Control Point	SSDP	Simple Service Discovery Protocol
DCP	Device Control Protocol	UDA	UPnP™ Device Architecture
DHCP	Dynamic Host Configuration Protocol	UPC	Universal Product Code
DNS	Domain Name System	URI	Uniform Resource Identifier
GENA	General Event Notification Architecture	URL	Uniform Resource Locator
HTML	HyperText Markup Language	URN	Uniform Resource Name
HTTP	Hypertext Transfer Protocol	UUID	Universally Unique Identifier
HTTPMU	HTTP (Multicast over UDP)	XML	Extensible Markup Language
HTTPU	HTTP (Unicast over UDP)		

References and resources

RFC 2616 — HTTP: Hypertext Transfer Protocol 1.1. <http://www.ietf.org/rfc/rfc2616.txt>.

RFC 2279 — UTF-8, a transformation format of ISO 10646 (character encoding). <http://www.ietf.org/rfc/rfc2279.txt>.

XML — Extensible Markup Language. W3C recommendation. <http://www.w3.org/TR/2000/REC-xml-20001006>.

Each clause in this document contains additional information about resources for specific topics.

0 Addressing

Addressing is Step 0 of UPnP™ networking. Through addressing, devices get a network address. Addressing enables discovery (Step 1) where control points find interesting device(s), description (Step 2) where where control points learn about device capabilities, control (Step 3) where a control point sends commands to device(s), eventing (Step 4) where control points listen to state changes in device(s), and presentation (Step 5) where control points display a user interface for device(s).

The foundation for UPnP networking is IP addressing. Each UPnP device which does not itself implement a DHCP server must have a Dynamic Host Configuration Protocol (DHCP) client and search for a DHCP server when the device is first connected to the network (if the device itself implements a DHCP server, it may allocate itself an address from the pool that it controls). If a DHCP server is available, i.e., the network is managed, the device must use the IP address assigned to it. If no DHCP server is available, i.e., the network is unmanaged; the device must use automatic IP addressing (Auto-IP) to obtain an address.

Auto-IP as defined herein is how a device: (a) determines if DHCP is unavailable, and (b) intelligently chooses an IP address from a set of link-local IP addresses. This method of address assignment enables a device to easily move between managed and unmanaged networks.

This clause provides a definition of the operation of Auto-IP. The operations described in this clause are detailed and clarified in the reference documents listed below. Where conflicts between this document and the reference documents exist, the reference document always takes precedence.

0.1 Addressing: Determining whether to use Auto-IP

A device that supports Auto-IP and is configured for dynamic address assignment begins by requesting an IP address via DHCP by sending out a DHCPDISCOVER message. The amount of time this DHCP Client should listen for DHCPOFFERs is implementation dependent. If a DHCPOFFER is received during this time, the device must continue the process of dynamic address assignment. If no valid DHCPOFFERs are received, the device may then auto-configure an IP address.

0.2 Addressing: Choosing an address

To auto-configure an IP address using Auto-IP, the device uses an implementation-dependent algorithm for choosing an address in the 169.254/16 range. The first and last 256 addresses in this range are reserved and must not be used.

The selected address must then be tested to determine if the address is already in use. If the address is in use by another device, another address must be chosen and tested, up to an implementation dependent number of retries. The address selection should be randomized to avoid collision when multiple devices are attempting to allocate addresses. It is recommended that the device choose an address using a pseudo-random algorithm (distributed over the entire address range from 169.254.1.0 to 169.254.254.255) to minimize the likelihood that devices that join the network at the same time will choose the same address and subsequently choose alternative addresses in the same sequence when collisions are detected. This pseudo-random algorithm may be seeded using the device's Ethernet hardware MAC address.

0.3 Addressing: Testing the address

To test the chosen address, the device must use an Address Resolution Protocol (ARP) probe. An ARP probe is an ARP request with the device hardware address used as the sender's hardware address and the sender's IP address set to 0s. The device will then listen for responses to the ARP probe, or other ARP probes for the same IP address. If either of these ARP packets is seen, the device must consider the address in use and try a different address. The ARP probe may be repeated for greater certainty that the address is not already in use; it is recommended that the probe be sent four times at two-second intervals.

After successfully configuring a link-local address, the device should send two gratuitous ARPs, spaced two seconds apart, this time filling in the sender IP address. The purpose of these gratuitous ARPs is to make sure that other hosts on the net do not have stale ARP cache entries left over from some other host that may previously have been using the same address.

Devices that are equipped with persistent storage may record the IP address they have selected and on the next boot use that address as their first candidate when probing, in order to increase the stability of addresses and reduce the need to resolve address conflicts.

Address collision detection should not be limited to the address testing phase, when the device is sending ARP probes and listening for replies. Address collision detection is an ongoing process that is in effect for as long as the device is using a link-local address. At any time, if a device receives an ARP packet with its own IP address given as the sender IP address, but a sender hardware address that does not match its own hardware address, then the device should treat this as an address collision and should respond as described in either a) or b) below:

a) Immediately configure a new link-local IP address as described above; or,

b) If the device currently has active TCP connections or other reasons to prefer to keep the same IP address, and has not seen any other conflicting ARP packets recently (e.g., within the last ten seconds) then it may elect to attempt to defend its address once, by recording the time that the conflicting ARP packet was received, and then broadcasting one single gratuitous ARP, giving its own IP and hardware addresses as the source addresses of the ARP. However, if another conflicting ARP packet is received within a short time after that (e.g., within ten seconds) then the device should immediately configure a new Auto-IP address as described above.

The device should respond to conflicting ARP packets as described in either a) or b) above; it should not ignore conflicting ARP packets. To switch over from one IP address to a new one, the device should, if possible, cancel any outstanding advertisements made on the previous address, and must issue new advertisements on the new address. The clause on Discovery explains advertisements and their cancellations.

After successfully configuring an Auto-IP address, all subsequent ARP packets (replies as well as requests) containing an Auto-IP source address should be sent using link-level *broadcast* instead of link-level *unicast*, in order to facilitate timely detection of duplicate addresses. As an alternative, a device which cannot send broadcast ARP replies should send a unicast ARP reply but then neglect to follow the instructions in RFC 826 about recording sender information from received ARP requests. This means that, having failed to record the sender information, the device is likely to send a broadcast ARP request of its own shortly later, which allows another device using the same IP address to detect the conflict and respond to it.

IP packets whose source or destination addresses are in the 169.254/16 range must not be sent to any router for forwarding. IP datagrams with a multicast destination address and an Auto-IP source address should not be forwarded off the local link. Devices and control points may assume that all 169.254/16 destination addresses are on-link and directly reachable. The 169.254/16 address range MUST NOT be subnetted.

0.4 Addressing: Periodic checking for dynamic address availability

A device that has auto-configured an IP address must periodically check for the existence of a DHCP server. This is accomplished by sending DHCPDISCOVER messages. How often this check is made is implementation dependent, but checking every 5 minutes would maintain a balance between network bandwidth required and connectivity maintenance. If a DHCPOFFER is received, the device must proceed with dynamic address allocation. Once a DHCP assigned address is in place, the device may release the auto-configured address, but may also choose to maintain this address for a period of time (or indefinitely) to maintain connectivity.

To switch over from one IP address to a new one, the device should, if possible, cancel any outstanding advertisements made on the previous address, and must issue new advertisements on the new address. The clause on Discovery explains advertisements and their cancellations.

0.5 Addressing: Device naming and DNS interaction

Once a device has a valid IP address for the network, it can be located and referenced on that network through that address. There may be situations where the end user needs to locate and identify a device. In these situations, a friendly name for the device is much easier for a human to use than an IP address. If a UPnP device chooses to provide a host name to a DHCP server and register with a DNS server, the device should either ensure the requested host name is unique or provide a means for the user to change the requested host name. Most often, UPnP devices do not provide a host name, but provide URLs using literal (numeric) IP addresses.

Moreover, names are much more static than IP addresses. Clients referring a device by name don't require any modification when the IP address of a device changes. Mapping of the device's DNS name to its IP address could be entered into DNS database manually or dynamically according to RFC 2136. While devices supporting dynamic DNS updates can register their DNS records directly in DNS, it is also possible to configure a DHCP server to register DNS records on behalf of these DHCP clients.

0.6 Addressing: Name to IP address resolution

A device that needs to contact another device identified by a DNS name needs to discover its IP address. The device submits a DNS query according to RFC1034 and 1035 to the pre-configured DNS server(s) and receives a response from a DNS server containing the IP address of the target device. A device can be statically pre-configured with the list of DNS servers. Alternatively a device could be configured with the list of DNS server through DHCP, or after the address assignment through a DHCPINFORM message.

0.7 Addressing references

RFC1034 — Domain Names - Concepts and Facilities. <http://www.ietf.org/rfc/rfc1034.txt>.

RFC1035 — Domain Names - Implementation and Specification. <http://www.ietf.org/rfc/rfc1035.txt>.

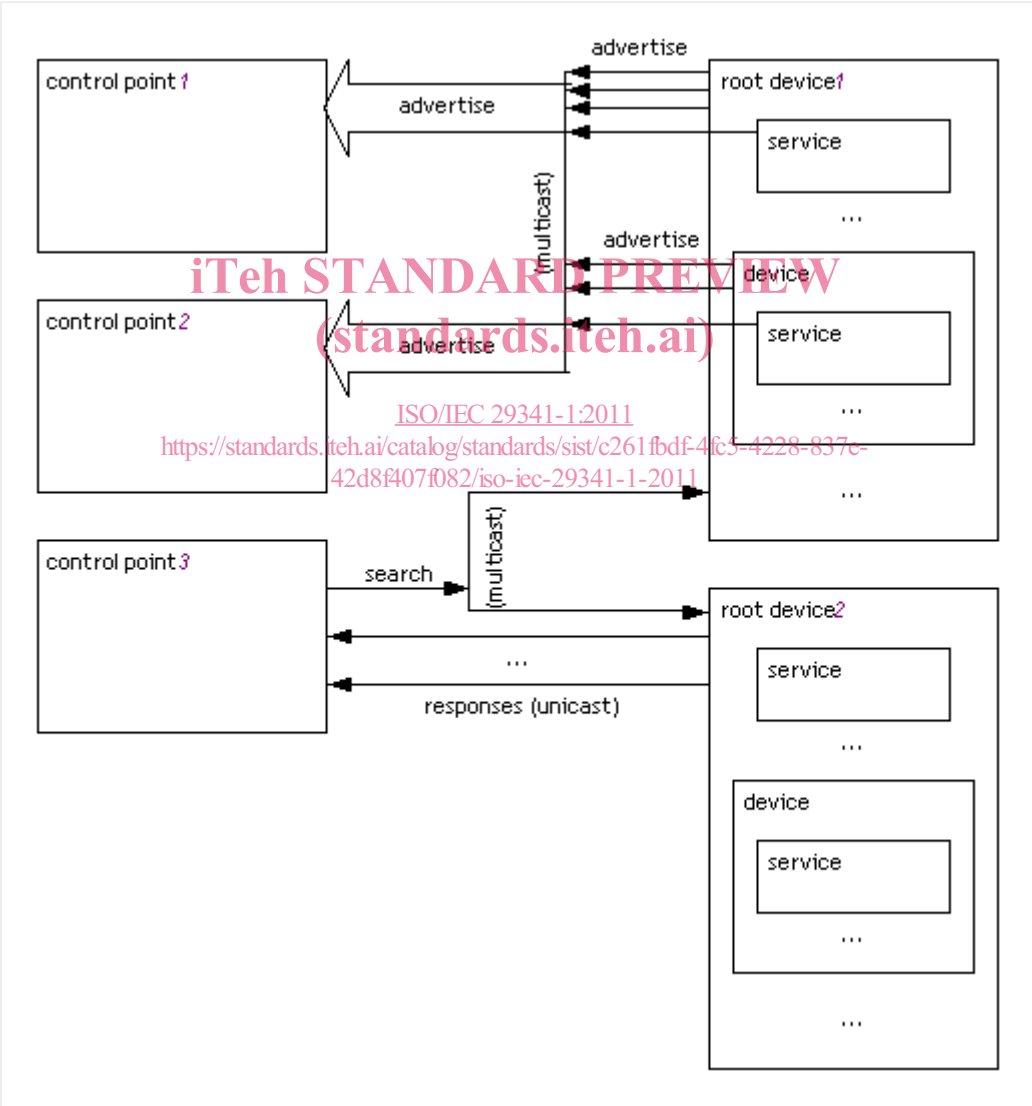
RFC 2131 — Dynamic Host Configuration Protocol. <http://www.ietf.org/rfc/rfc2131.txt>.

RFC 2136 — Dynamic Updates in the Domain Name System. <http://www.ietf.org/rfc/rfc2136.txt>.

1 Discovery

Discovery is Step 1 in UPnP™ networking. Discovery comes after addressing (Step 0) where devices get a network address. Through discovery, control points find interesting device(s). Discovery enables description (Step 2) where control points learn about device capabilities, control (Step 3) where a control point sends commands to device(s), eventing (Step 4) where control points listen to state changes in device(s), and presentation (Step 5) where control points display a user interface for device(s).

Discovery is the first step in UPnP networking. When a device is added to the network, the UPnP discovery protocol allows that device to advertise its services to control points on the network. Similarly, when a control point is added to the network, the UPnP discovery protocol allows that control point to search for devices of interest on the network. The fundamental exchange in both cases is a discovery message containing a few, essential specifics about the device or one of its services, e.g., its type, universally unique identifier, and a pointer to more detailed information.



When a new device is added to the network, it multicasts a number of discovery messages advertising itself, its embedded devices, and its services. Any interested control point can listen to the standard multicast address for notifications that new capabilities are available.

Similarly, when a new control point is added to the network, it multicasts a discovery message searching for interesting devices, services, or both. All devices must listen to the standard multicast address for these messages and must respond if any of their embedded devices or services match the search criteria in the discovery message.