

A faint, light blue flowchart is visible in the background of the cover. It consists of numerous rectangular and diamond-shaped nodes connected by thin white arrows, creating a complex web of paths that suggest a process or data flow.

RDMS 202
RDMS 202D
Users Manual
Version 1.01

August 2010

Allegro Software Development Corporation

1740 Massachusetts Avenue

Boxborough, MA 01719

Telephone: 978 264-6600 Fax: 978 266-2839

www.allegrosoft.com



COPYRIGHTS

This manual is © Copyright 1996-2010 by Allegro Software Development Corporation. All rights reserved. RomPager®, RomXML®, RomPlug® and RomXOAP® are trademarks registered in the U.S. Patent and Trademark Office to Allegro Software Development Corporation. RomCLI, RomDNS, RomMailer, RomPOP, RomRadius, RomSShell, RomTime, RomWebClient, and Allegro Media Server are trademarks of Allegro Software Development Corporation. The RomPager and RomMailer products described in this manual are protected by U.S. Patent No. 6,782,427. UPnP™ is a certification mark of the UPnP Implementers Corporation. The name of other companies and products mentioned herein are the trademarks of their respective owners.

TABLE OF CONTENTS

Copyrights	ii
Table of Contents	iii
Preface	v
Preface	v
CHAPTER 1 Introduction	1
RomPager Product Family	1
CHAPTER 2	
UPNP / DLNA Overview	4
UPnP™/DLNA® Overview	4
<i>Evolution of DLNA</i>	5
<i>DLNA Device Model</i>	5
<i>DLNA Device Categories and Classes</i>	6
DLNA Architecture	12
<i>DLNA Media Formats</i>	12
<i>DLNA Device Discovery, Description, Control, Eventing and Media Management</i>	13
<i>Device Discovery</i>	13
<i>Description</i>	13
<i>Control</i>	13
<i>Eventing</i>	14
<i>Presentation</i>	14
<i>Media Management</i>	14
<i>Content Directory</i>	14
<i>Connection Manager</i>	15
<i>AV Transport</i>	15
<i>Rendering Control</i>	15
<i>DLNA Media Transport</i>	16
<i>Network Stack</i>	16
<i>Network Connectivity</i>	16
Link Protection and Digital Rights Management (DRM)	16
Certification	17
Chapter 3	
RDMS 202 Getting Started	18
RDMS 202 - Getting Started	18
RDMS 202 - Getting to Know the RDMS 202	18
Applicable Documents	21
RDMS 202 Requirements	21
Connecting the RDM2 202 or RDMS 202D to a Network	22

Chapter 4
RDMS 202 Operation

	23
Interacting with the RDMS 202	23
Preferences	26
Status	31
Content	33
Add Content	35

Chapter 5
Media Formats, CDS and USB Drive

	36
Understanding the RDMS Content Directory Service	36
PQI Directory Structure	38
Reset to Factory Defaults	39
Adding Content to RDMS 202 or RDMS 202D	40
Merge Upload	41
Deleting Content from the RDMS 202 or RDMS 202D	41
<i>Deleting a single item</i>	41
<i>Delete Uploads</i>	42

Preface

Thank you for purchasing the Allegro Reference Digital Media Server (RDMS 202/RDMS 202D). The RDMS 202 is intended as a testing and development tool for OEMs producing DLNA enabled products. The RDMS 202 meets all DLNA 1.5 interface guidelines as a Digital Media Server (DMS). As delivered, the RDMS 202 is provided with DLNA Certification test content for audio, video, and images. The RDMS 202 is available to DLNA Member companies that have signed the DLNA Content Agreement document. The RDMS 202D additionally supports link protection utilizing DTCP-IP and is available to DLNA Member companies that also have a valid DTLA license agreement.

The following text provides valuable information for the integration and use of Allegro products with your embedded development project. To better understand the layout of information within the manual, a brief summary of the organization and contents are provided below.

Chapter 1 provides an overview and high level introduction to the full RomPager Suite of Internet Software Development Toolkits offered by Allegro Software Development Corporation. This provides a high level overview of how each of the individual products are related and potentially used in concert with other established Allegro embedded technology.

Chapter 2 gives an overview of Connected Consumer Electronics and how the Digital Lifestyle Networking Alliance (DLNA) has brought together a set of standards based protocols to ensure wide spread and seamless interoperability.

Chapter 3 familiarizes the user with the RDMS 202 / RDMS 202D and how to initially setup RDMS in a DLNA environment.

Chapter 4 walks the user through the details of operating the RDMS in a DLNA environment including example screen shots for all user interface pages.

Chapter 5 takes a closer look at the supplied USB drive and the pre-provisioned DLNA testing content.

While the manual does include a great deal of detail about the RDMS and how to utilize the products within a DLNA testing environment, it is unlikely to cover every possible use case scenario. If you need to reach our support staff with questions regarding the product, please visit us online to log your support request. An Allegro Support representative will follow-up with your request shortly.

http://www.allegrosoft.com/support_req.html

Again, Thank you for purchasing the RDMS 202 / RDMS 202D from Allegro Software Development Corporation.

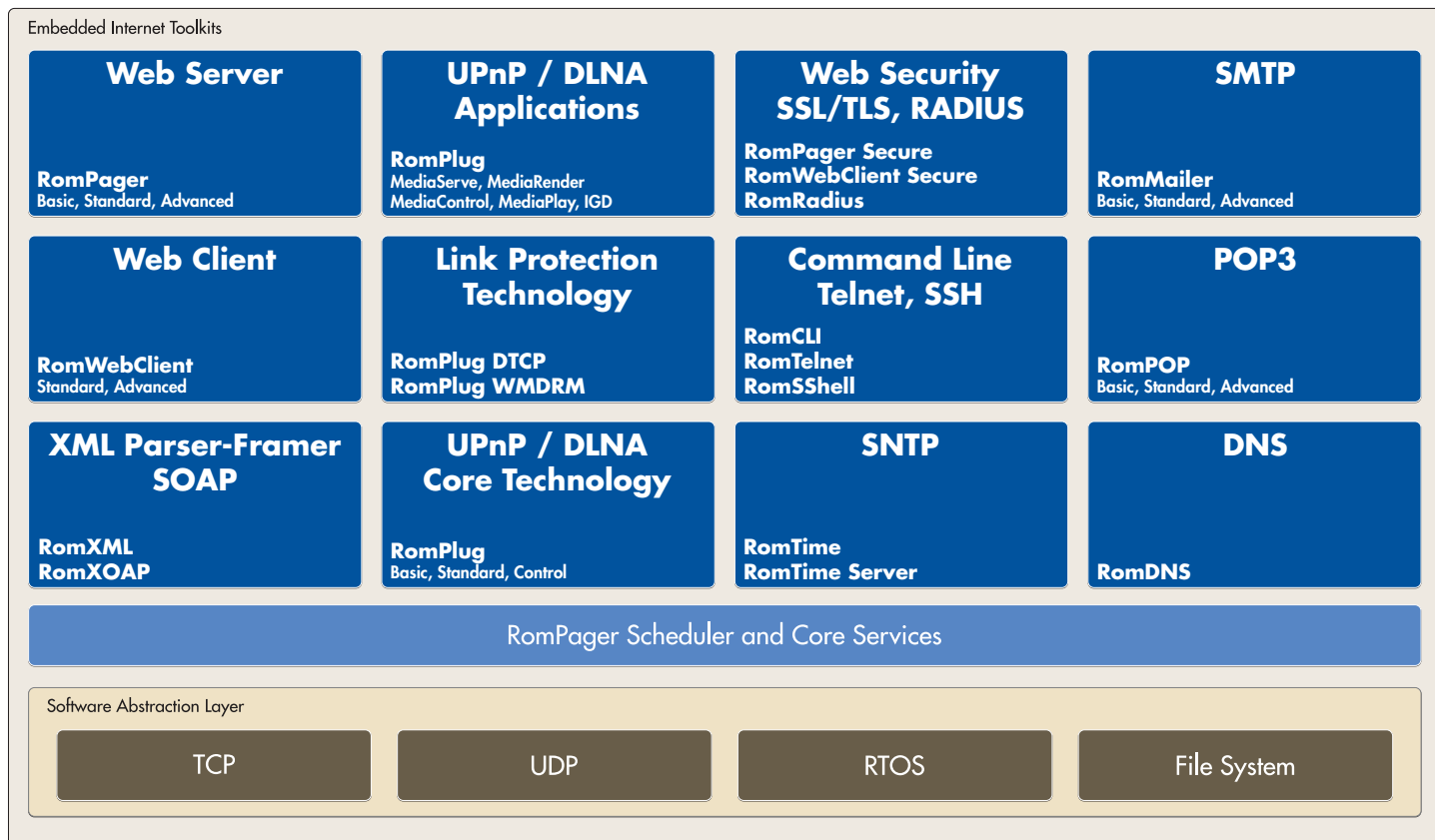
CHAPTER 1

INTRODUCTION

RomPager Product Family

The RomPager® family of products bring Web, email and other Internet services to embedded devices. The RomPager toolkits allow virtually any device to use standard Internet applications. A device can use Internet standard protocols to serve pages, images or applets; retrieve files from Web servers; and send or receive email with attachments; easily allowing Internet-based man-machine interaction. The RomPager products also allow embedded device designers to easily develop machine-to-machine systems using standard Internet techniques such as Web Services or SOAP.

The RomPager product family is provided in ANSI-C source code and has been ported to all major processor/OS platforms and is delivered with interface files for the leading OS environments. The RomPager products use a common Software Abstraction Layer to provide an interface with any RTOS, TCP/IP and file system environment. With a typical OS, the RomPager product family shares a single task/thread in the device environment using a common light-weight scheduler to support multiple simultaneous HTTP and other protocol requests. In fact, the Allegro products can run in devices without an OS, by running off the idle loop. All of the protocol products will run separately or in combination with the others. Sophisticated compiler option flags allow maximum code-sharing to provide the smallest possible code footprint.



There are three RomPager Web Server toolkits that provide a range of embedded Web server capabilities. The **RomPager Basic** toolkit is an HTTP 1.0/1.1 Web server with CGI-style user exit support and optional file system support. It uses from 7Kb to 12Kb of ROM code and provides a small, powerful server for low-end devices. The **RomPager Standard** toolkit includes all the capabilities of the RomPager Basic toolkit and adds the PageLoader web object compiler. PageLoader imports Web pages (prepared with any Web page layout program), applets and graphics such as GIF, JPG and PNG and creates a compressed Web object library for the device. The **RomPager Advanced** toolkit includes all the capabilities of the RomPager Standard toolkit and provides additional HTTP 1.0/1.1 features, an internal security database and the PageBuilder compiler. The PageBuilder compiler provides the features of PageLoader and a comprehensive Web Application development environment including full support for HTML (2.0, 3.2, and 4.0), XHTML, Javascript, object compression, application compression and support for internationalization with dynamic phrase dictionaries.

Optional packages for the RomPager Advanced Web server include **SoftPages**, **Remote Host** and **Graphlets**. SoftPages adds a runtime HTML parser to RomPager and allows the device vendor to make runtime source changes to HTML pages. Remote Host provides integrated HTTP proxy services to support redirection of HTTP requests from the RomPager Web server to another Web server for retrieving objects too large to store in the embedded device. The Graphlets toolkit is a series of Java applets that provide graphic control indicators for the embedded device. The applets include line charts, bar charts, progress bars and dial indicators.

RomWebClient™ Standard is an HTTP 1.0/1.1 client that provides embedded devices the ability to retrieve and store objects from remoteWeb servers using the HTTP protocol. Objects can be in any format and can be used in a memory buffer or stored in the optional file system. **RomWebClient Advanced** adds caching, cookies and pipelining capabilities. The Web clients interoperate with any standard Web server or with other embedded devices that have embedded Web servers.

The **RomPager Secure** and **RomWebClient Secure** toolkits provide SSL 3.0 and TLS 1.0, 1.1 secure server and client sessions. The encryption protocols interoperate with any secure browser or server and include RSA, RC4 DES, 3DES, SHA and AES algorithms. The RomWebPager and RomWebClient secure toolkits offer standalone security capabilities or are available as integrated options for the RomPager server or RomWebClient toolkits.

RomPlug® Basic is a toolkit for implementing the Device Discovery and Description sections of the UPnP™ architecture. The **RomPlug Advanced** Device toolkit adds support for the Control and Eventing sections of the UPnP architecture to build certified UPnP devices. The **RomPlug Control** toolkit provides the ability to build fully compliant UPnP architecture Control Points. Optional application toolkits provide support for **UPnP IGD**, **Media Renderer**, **Media Server**, **Media Player** and **Media Controller** devices. The **RomPlug DTCP** and **RomPlug WMDRM** toolkits allow engineering teams to easily integrate link protection into state of the art UPnP and DLNA enabled consumer electronics and mobile devices.

RomCLI™ is a toolkit that may be used to build Command Line Interfaces similar to Cisco IOS-based products. The RomCLI toolkit includes an offline compiler (CliBuilder) for preparing command definitions and also includes a built-in Telnet server as well as serial support. The **RomSShell™** toolkit provides a SSH Version 2 server that may be used by itself or as an integrated front end to RomCLI.

RomMailer™ Basic is a Simple Mail Transfer Protocol (SMTP) client that enables embedded devices to send Internet email text messages. **RomMailer Standard** adds support for attachment files using MIME and UUENCODED formats. **RomMailer Advanced** provides Delivery Status Notification and Message Delivery Notification support. When RomMailer is used with the RomPager Advanced Server, messages can also be HTML Mail with embedded graphics and dynamic insertion of variables into the message text.

RomPOP™ Basic is a toolkit for building a Post Office Protocol (POP3) client so that embedded devices can receive Internet email text messages. **RomPOP Standard** adds support for attachment files using MIME and UUENCODED formats. **RomPOP Advanced** provides Delivery Status Notification and Message Delivery Notification support.

RomDNS™ is a Domain Name Services client that provides embedded devices the ability to perform lookups of a variety of DNS records. It may be used to simplify configuration for RomMailer and RomPOP, or to provide server addresses for RomWebClient.

RomTime™ is a Network Time Protocol (NTP) client that provides embedded devices the ability to receive time services from a network time server, allowing devices to avoid requiring the user to enter the time manually.

RomTime Server is an NTP server designed specifically for use in embedded applications.

The **RomXML®** toolkit is a small eXtensible Markup Language (XML) implementation that enables your embedded device to send (frame) and receive (parse) XML documents. Using XML in your embedded designs provides for free-format interchange of data and is widely accepted in the device management, remote sensing and enterprise IT communities. Allegro's RomXML has been designed from the ground up for use in embedded devices that often have limited resources. Written in ANSI-C, the toolkit offers built in capabilities to convert internal data between C language structures and XML documents. The **RomXoap®** toolkit builds upon the capabilities of RomXML and offers design engineers a comprehensive solution for creating connectivity between embedded designs and enterprise IT environments utilizing standards based SOAP technology. Available as standalone toolkits or tightly integrated with the other RomPager family of products, RomXML and RomXoap provide the foundation for enabling embedded devices with XML, SOAP and Web Services capabilities

CHAPTER 2

UPNP / DLNA OVERVIEW

UPnP™/DLNA® Overview

Consumers have embraced digital technologies and given rise to the digital home. Over the past few years consumers have started to acquire, enjoy and manage an ever increasing amount of digital content. A large library of digital music, various photos from digital cameras, cell phones, home videos and more recently premium video content downloaded from the Internet are just a few examples. The scale, options and size of digital content will only grow over the coming years. Along with the shift to digital content, consumers want an environment that allows all their devices in the home to easily work together regardless of vendor. This vision of the digital home integrates Personal Computers (PC), Consumer Electronics (CE) and Mobile Devices (MD) with a seamless interoperable network. It is also one that requires a common set of industry design guidelines to allow companies to build pervasive interoperability.

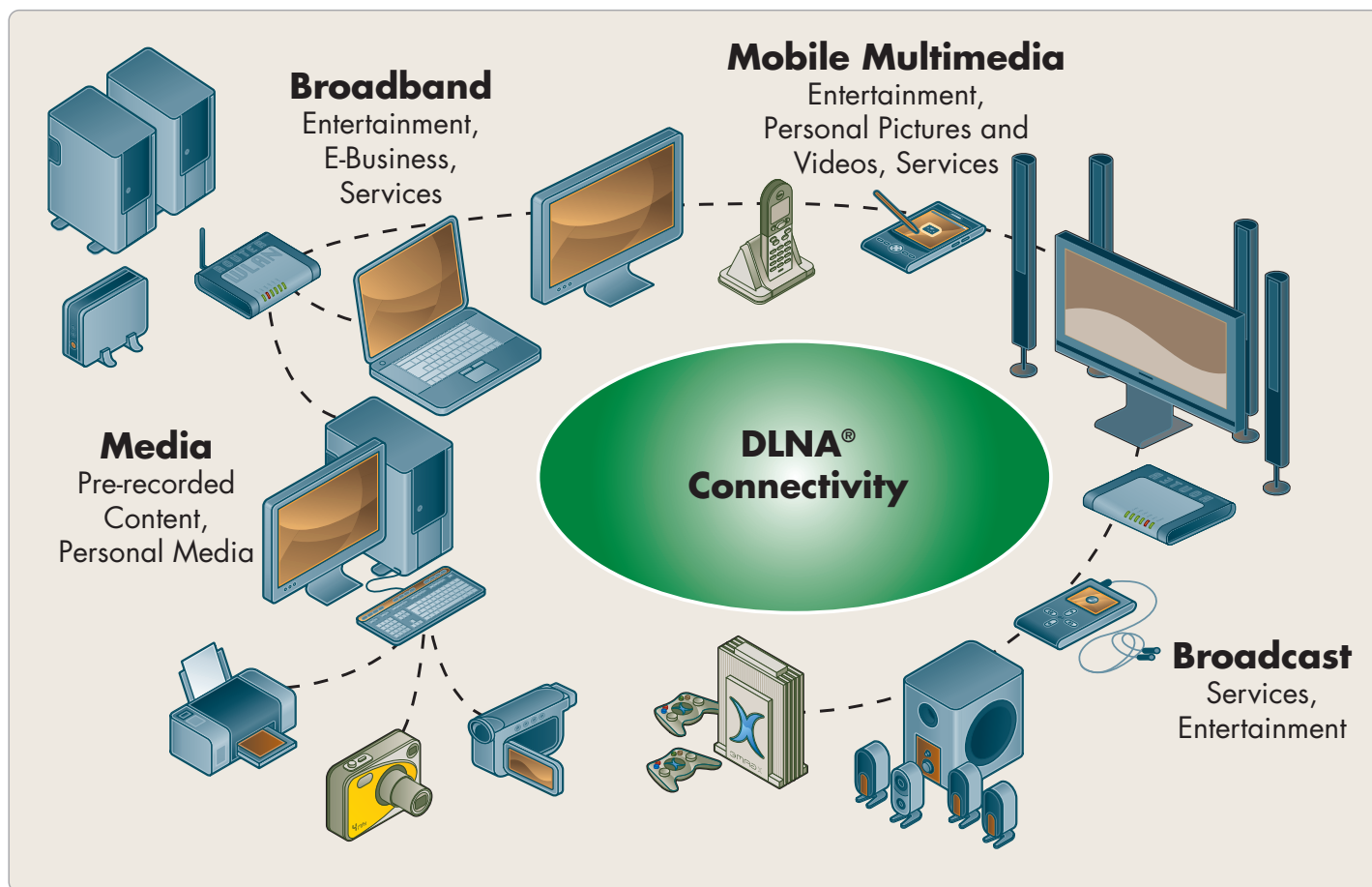


Figure 2.1 - Interoperable DLNA home network example

Evolution of DLNA

The Digital Living Network Alliance (DLNA) was formed to address the need for a common set of industry design guidelines. The DLNA Home Networked Device Interoperability Guidelines were created by a unique cross-industry effort that combined the efforts of over 250 Consumer Electronics, PC and Mobile Device companies from around the world who worked together with the aim of achieving the world's first substantial approach to true interoperability between personal computers, consumer electronics, and mobile devices. The Interoperability Guidelines provide product developers with a long-term architectural view, plus specific guidance for IP-networked platforms, devices and applications in the home.



The ability of DLNA to deliver workable interoperability guidelines in less than 12 months is largely due to the pioneering efforts of the Universal Plug and Play Forum (UPnP Forum - www.upnp.org). The mission of the Forum is simple: interoperability between devices using industry standards. To that end, the Forum selected TCP/IP as the basis for all network connectivity. Added to TCP/IP were Web standards such as HTTP, HTML, XML, and SOAP that provided the framework for device discovery, device and services description, control, and presentation.



With the core architecture defined, the Forum established a series of working groups to define device and service profiles for specific device categories. These categories include Audio/Video (AV), Internet Gateway Device (IGD), Printing, Scanning, Lighting Control, HVAC, and a number of others. The working groups – composed of member companies from relevant industries – delivered a series of XML schemas representing the baseline set of functions and services that each specific device type was required to support.

The most significant of the working groups efforts (at least in terms of digital media content) was the UPnP AV specification. In fact, it was so important that it became the basis for a new organization – DLNA. DLNA was formed in 2003 by 21 companies including Microsoft, Intel, HP, IBM, Sony, Philips, Toshiba, Pioneer, Motorola and Nokia, with the goal of accelerating the development and deployment of interoperable digital media devices for the home.

DLNA Device Model

The device model used by DLNA is derived from the UPnP Forum fundamental device model. This model consists of Devices, Services, and Control Points.

Devices are network entities that provide services and can contain other nested devices.

Services are the basic unit of control. They provide actions, and maintain status via state variables.

Control Points are network entities that are capable of discovering and controlling other devices on the network.

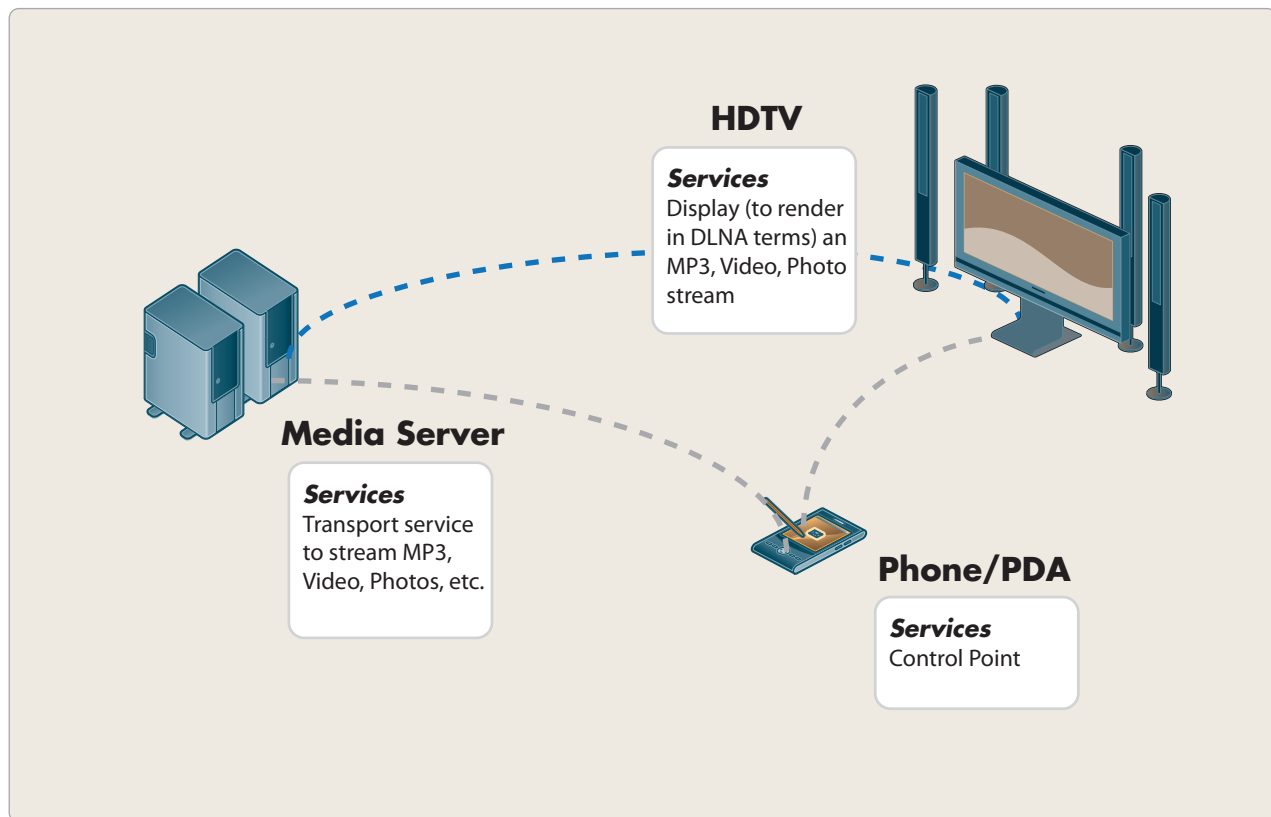


Figure 2.2 - DLNA devices interacting in a typical home environment.

In many cases it is very common for a DLNA *Device* to consist of multiple *Services* and potentially contain a *Control Point* as well. In Figure 2.2 above, the Media Server *Device* provides a transport service for streaming audio MP3 files, photos and movies stored on the internal RAID disks. The HDTV is a *Device* that provides a *Service* to display video content on the screen (in UPnP / DLNA terms it will render the content to the screen). The Phone / PDA is a *Control Point* used to tell the Media Server to stream stored content to the HDTV to display (or render) to the screen.

With the development of the UPnP AV (and thus DLNA) specifications for digital media content devices, the basic device model was extended. All control interaction (shown as dotted gray lines) only passes between a *Control Point* and *Device(s)*, but the *Devices* themselves interact (shown as dotted blue lines) with each other to pass digital content using a non-UPnP (“out-of-band”) communications protocol.

DLNA Device Categories and Classes

To better define the characteristics of devices and the services they offer, the DLNA Interoperability Guidelines define three *Device Categories*:

- Home Network Devices (HND)
- Mobile Network Devices (MND)
- Home Infrastructure Devices (HID)

Device Categories are based on a shared set of media formats and network connections with the focus on interoperability between devices within a category. *Devices* can and often do belong to more than one *Device Category*. The underlying requirement is that the *Device* must comply with the media formats and network connectivity of both categories.

Each *Device Category* is further broken into *Device Classes*. A *Device Class* specifies the functional capabilities of a device regardless of its physical attributes. In fact, a single physical device can, and frequently does incorporate multiple *Device Classes*. For interoperability, DLNA performs device certification at the *Device Class* level. All DLNA Certified™ devices must comply with all the requirements of the *Device Class(es)* that they belong.

The Home Network Device (HND) category is made up of five *Device Classes* that are in use in the home network, and rely on the same media formats and network connectivity requirements.

- Digital Media Server (DMS)
- Digital Media Renderer (DMR)
- Digital Media Controller (DMC)
- Digital Media Printer (DMP_r)
- Digital Media Player (DMP)

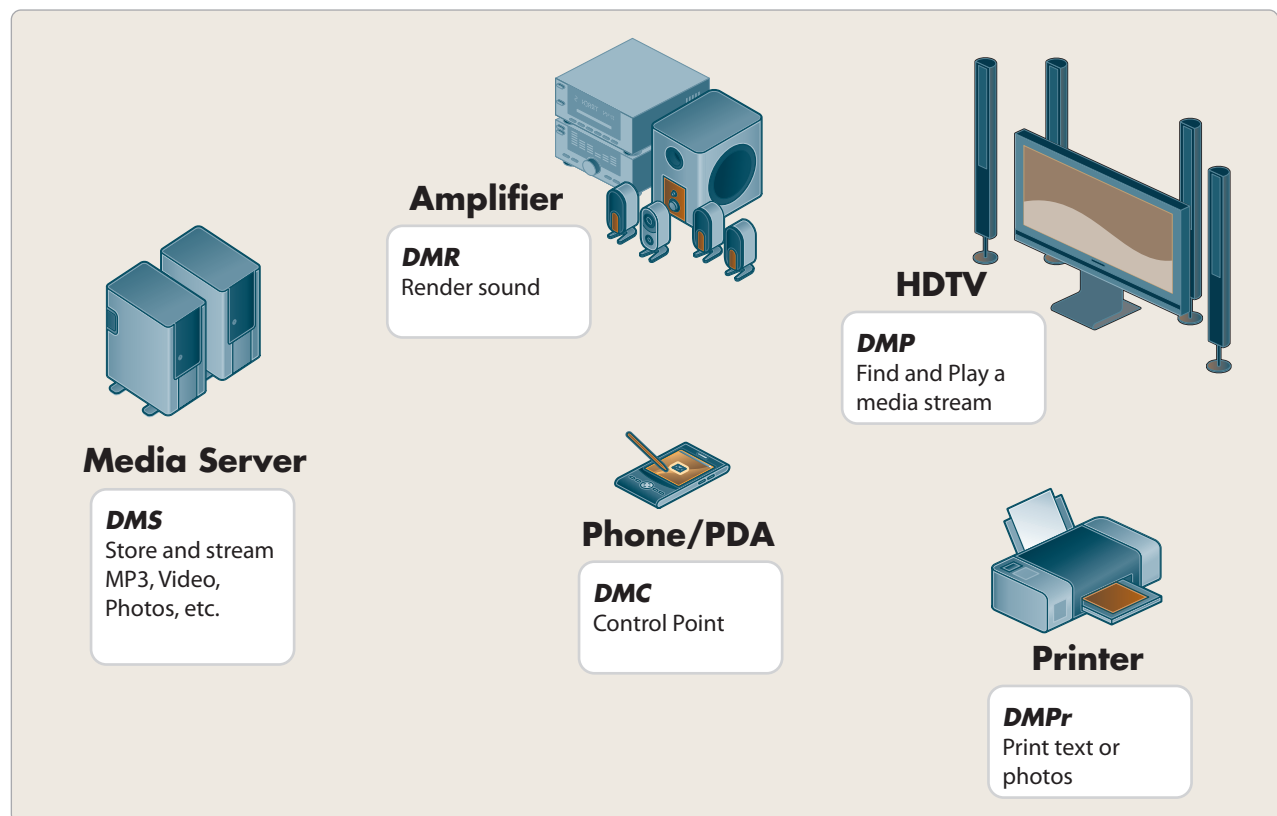


Figure 2.3 - DLNA Home Networking Device examples

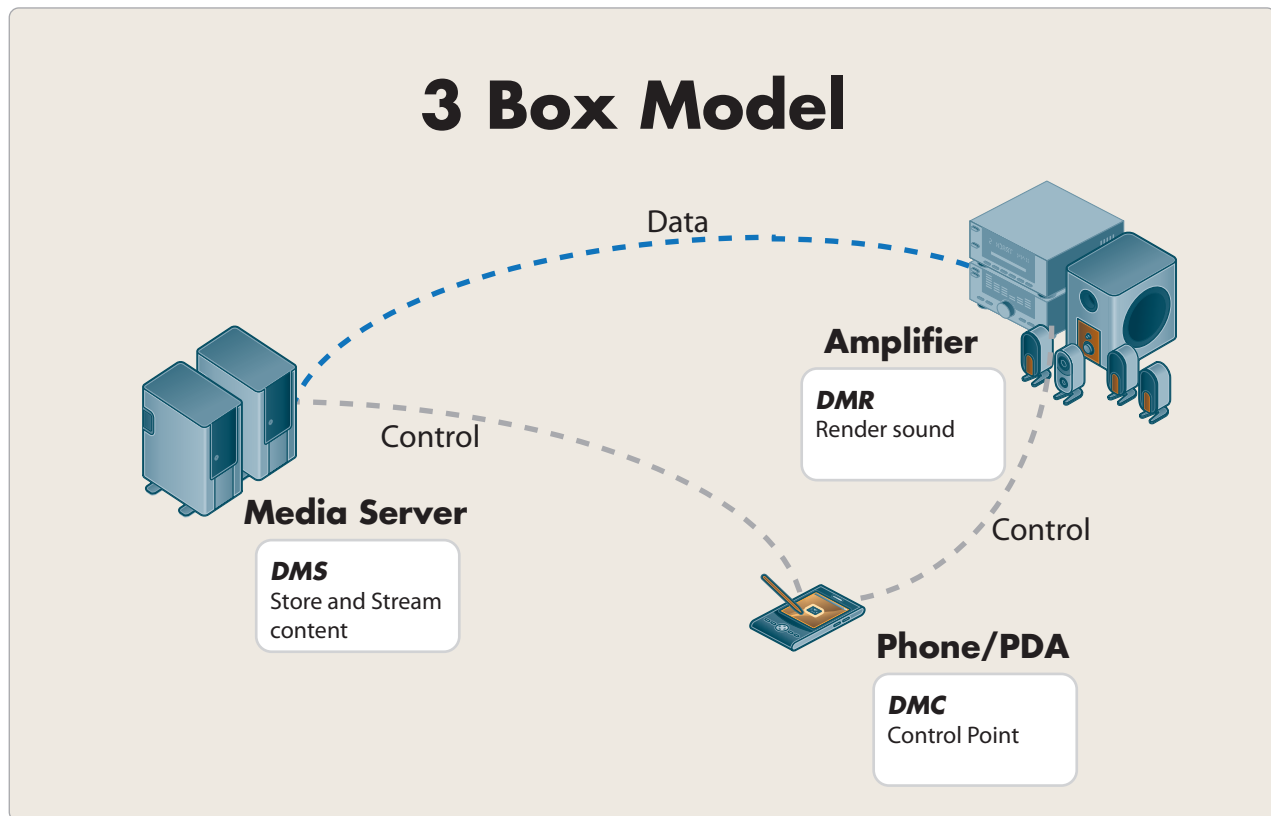


Figure 2.4 - DLNA 3 Box Model with a DMS, DMC and DMR

In the 3 Box Model (Figure 2.4) a consumer utilizes a DMC to discover DLNA devices on a home network. The DMC is used to browse and select content via a user interface (UI). The consumer continues by means of the DMC to select where the content will be played. Then, with the standards based DLNA architecture, the devices automatically connect and the content is rendered (played) for the user. The five device classes within the HND category bring this about, however there are a few characteristics worth noting.

In an effort to initially simplify DLNA architecture, two specific models or scenarios were defined and utilized by manufacturers of DLNA Certified products. The use of these scenarios bring to light various details about a DMP, DMR and DMC and how they interact in a DLNA home networking environment.

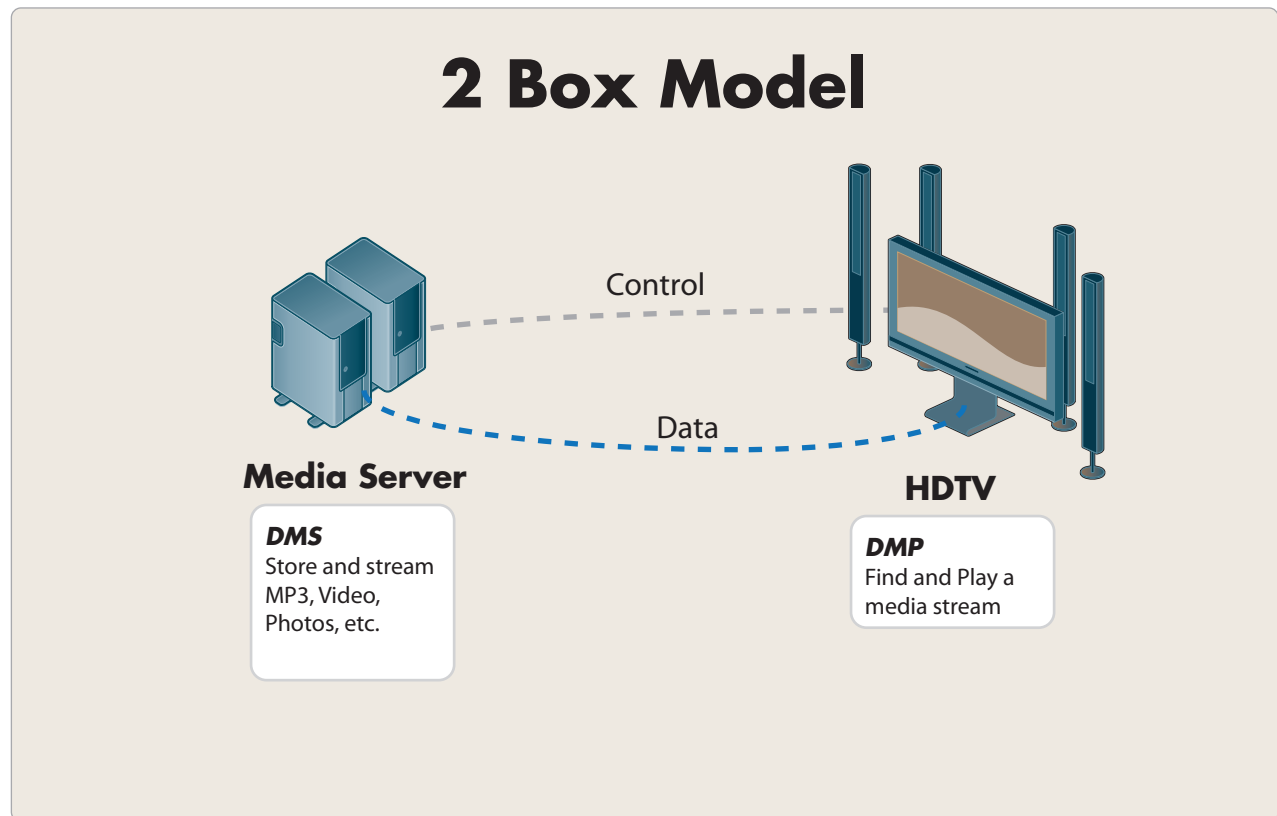


Figure 2.5 - DLNA 2 Box Model with a DMS and DMP

Figure 2.5 illustrates the DLNA 2 Box Model. The 2 Box model can be thought of as a special case of the 3 box implementation. In this use scenario, an HDTV acting as a DMP provides a consumer with an elegant user interface (UI) to find, browse, select and eventually view digital media. A DMP can be thought of as a combination of a DMC and DMR with a few additional characteristics. The DMC provides the control interface for finding, browsing and selecting content. The DMR provides the ability to render (or play) content to the wide screen TV. In the 3 Box model a DMC can discover and control (or cause) content to be rendered (or played) from any DMS to any DMR. In the case of a DMP and 2 Box Model, the DMC part of the DMP always selects the local DMR as the target renderer for all content. In addition the DMR in the 3 Box Model is always discoverable by other DLNA devices. The DMR within a DMP in the 2 Box Model is not discoverable (although some vendors also allow this DMR to be recognized outside of the DMP entity).

The Mobile Handheld Device (MHD) category is made up of five Device Classes that share the same usages models as the HND Device Category, but have different requirements for media format and network connectivity.

- Mobile Digital Media Server (M-DMS)
- Mobile Digital Media Controller (M-DMC)
- Mobile Digital Media Player (M-DMP)
- Mobile Digital Media Uploader (M-DMU)
- Mobile Digital Media Downloader (M-DMD)

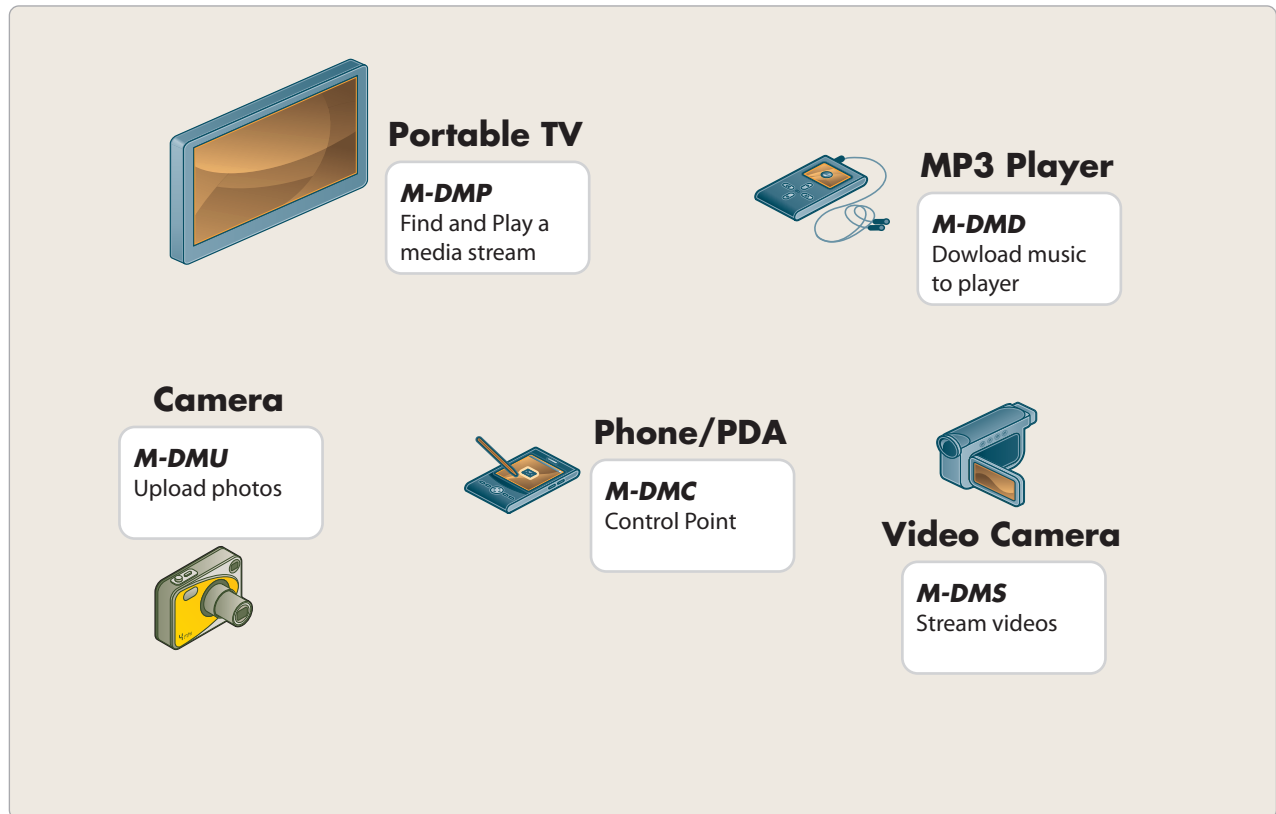


Figure 2.6 - DLNA Mobile Handheld Devices

Figure 2.6 illustrates the types of DLNA devices found within the Mobile Handheld Device category. It is important to note that just as the DMP could not be discovered in the HND the M-DMP Device Class is also undiscoverable by other DLNA devices.

The Home Infrastructure Device (HID) category is made up of two Device Classes. These devices are intended to allow HNDs and MHDs to interoperate.

- Mobile Network Connectivity Function (M-NCF)
- Media Interoperability Unit (MIU)

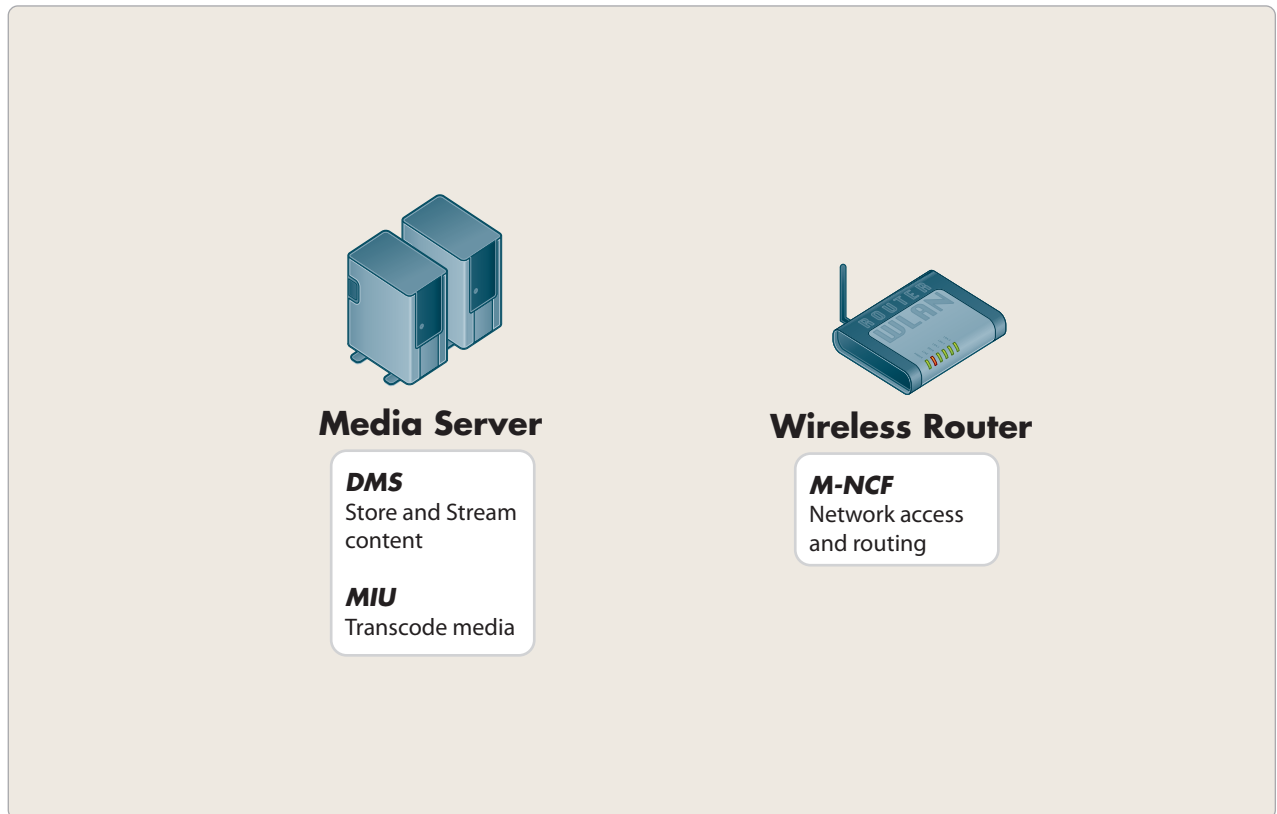


Figure 2.7- DLNA Home Infrastructure Device examples

Figure 2.7 illustrates examples of HID Devices in a home network. The Media Server is a member of the HND category as a DMS Device Class, this example shows the Media Server is also a member of the HID Device Category as a MIU Device Class. The MIU Device class provides a vital role in transcoding stored digital content into formats that mobile devices can consume. The M-NCF provides necessary access, routine and bridging functions to seamlessly connect mobile devices to a home network.

DLNA Architecture

Figure 2.8 illustrates the functional components of DLNA 1.5 as it relates to the Interoperability Guidelines network architecture. Each of the functional components are briefly discussed in the following sub-sections.

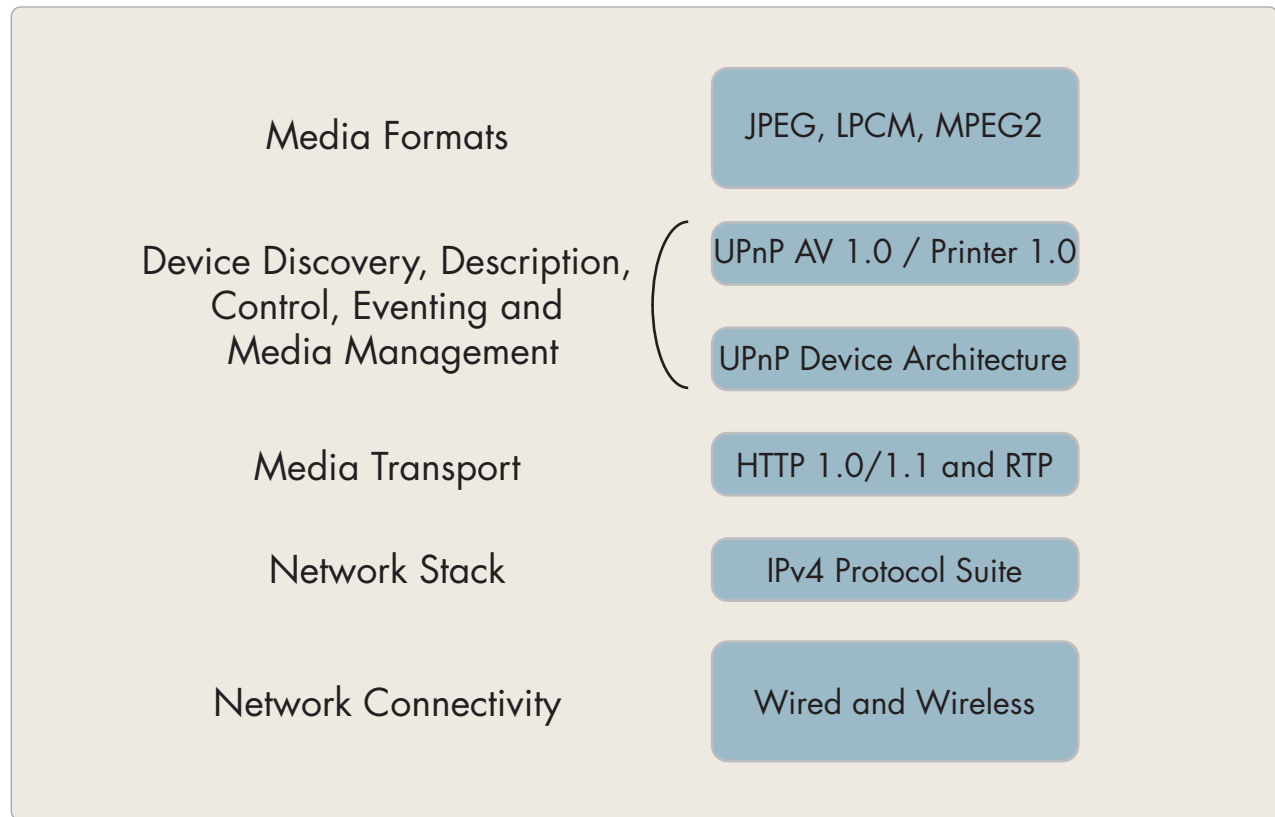


Figure 2.8 - DLNA Functional Component Architecture

DLNA Media Formats

Media Formats describe how digital content is encoded and formatted for each of the three *Media Classes*:

- Image
- Audio
- Audio Visual

Media format profiles are very explicit, with attributes, parameters, system, and compression level details defined in sufficient detail to ensure interoperability between DLNA Certified devices. The present set of *Media Formats* that must be supported by each specific device in a *Media Class* are listed in the Table 2.1. Optional media formats are also defined. The *Interoperability Guidelines* provide specific rules about using optional formats between compatible devices in addition to conversion between optional and mandatory formats.

Table 2.1 - Media Class mandatory and optional formats

Media Class	Mandatory Formats	Optional Formats
<i>Image</i>	JPEG	PNG, GIF, TIFF
<i>Audio</i>	LPCM	AC3, AAC, MP3, WMA9, ATRAC3plus
<i>Video</i>	MPEG2	MPEG1, MPEG4, VC1, MPV1

The focus on *Media Formats* is a key distinguishing factor between the UPnP Forum and DLNA. The UPnP Forum focused on achieving device interoperability, which was accomplished. But the lack of prescribed media profiles prevented the UPnP architecture from delivering media interoperability and led to the founding of DLNA.

DLNA Device Discovery, Description, Control, Eventing and Media Management

Device Discovery, Description, Control and *Eventing* enable a device on the home network to discover the presence and capabilities of other devices on the network and collaborate with these devices in a uniform and consistent manner. DLNA incorporates the UPnP Forum Device Architecture 1.0 as the basis for its device discovery and control.

Device 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 the control point to search for devices of interest on the network. The message exchanged in both cases is a discovery message containing a few, essential specifics about the device, its services (such as the device type), an identifier, and an HTTP URL to access for more detailed information. The UPnP discovery protocol is based on the Simple Service Discovery Protocol (SSDP).

Description

After a control point has discovered a device, the control point needs to find out more 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* using the URL provided by the device in the discovery message. This URL points to a UPnP description document that is expressed in XML and includes vendor-specific information such as the model name and number, serial number, manufacturer name, URLs to vendor-specific web sites and URLs for control, eventing, and presentation.

The UPnP description document includes a list of *Device Services* provided by the device. For each *Device Service*, the *Description* includes a list of actions for the service, and arguments for each. The *Description* of a *Device Service* also includes a list of variables that reflect the state of the device. These variables are described in terms of their data type, range, and event characteristics.

Control

After a control point has retrieved the *Description* of the device, the control point can invoke actions that are supported by the device. To do this, a control point sends a suitable control message to the control URL for the service. Control messages are expressed in XML using the Simple Object Access Protocol (SOAP). Control actions are like function calls, and the device service will return action-specific values in response to the control message. The effects of the action may also be reflected by changes in the variables that describe the run-time state of the device.

Eventing

The UPnP description document for a service includes a list of variables that represent the state of the device. The device publishes updates whenever there is a change in the value of any evented variable. A control point may subscribe to receive this information by sending a GENA (General Event Notification Architecture) message. The device publishes updates by sending event messages. Event messages contain the names of one or more state variables and their current values. These messages are expressed in XML and sent using HTTP.

A special initial event message is sent when a control point first subscribes. This initial 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 state of the device. This means that all subscribers are sent all event messages, and subscribers receive event messages for all evented variables that have changed.

Presentation

The presentation capabilities of a device enable HTML-based management of the device by the end user using a standard web browser. A control point can obtain the entry presentation URL from the device description document, retrieve the entry page from the URL, load the page into a browser, and start the user management of the device.

Media Management

Media Management enables devices and applications to identify, manage, and distribute digital media content across network devices. The *Interoperability Guidelines* incorporate the UPnP Forum AV technology as the basis for DLNA *Media Management*. There are four *Device Services* provided by this technology:

- Content Directory
- Connection Manager
- AV Transport
- Rendering Control

Content Directory

The *Content Directory* service provides a mechanism for each content server on the network to provide a uniform directory of all its available content to any interested devices on the network. Every content server must have an instance of this service.

This service might provide a list of songs stored on an MP3 player, still-images comprising various slide shows stored on a PC, movies stored in a DVD jukebox, TV shows currently being broadcast by a Set-top Box, songs stored in a media server, TV programs that had been downloaded to a PVR, photos stored in a digital camera, and many more. Nearly any type of content can be listed via the *Content Directory* service, even for devices that support multiple types of content. The information about the content (metadata) returned by the *Content Directory* service includes properties such as its name, artist, creation date, size, etc. In addition, the metadata also indicates the transfer protocols and data formats that are supported for each piece of content on the server. This information is used by the *Control Point* to determine if a given Media Renderer is capable of rendering the content in its current format or if some type of transcoding is required.

Connection Manager

The Connection Manager service determines how the digital media content can be transferred between two devices on the network. Each device that sends or receives content must implement the *Connection Manager* service. This service provides a mechanism for devices to:

- To determine whether a content item can be played on a specific device
- Set up and tear down connections between devices
- Discover information about current transfers in the network

AV Transport

The *AV Transport* service enables control over the “playback” of audio and video streams including the ability to Stop, Pause, Seek, etc. This service type defines a common model for *AV Transport* control suitable for a generic user interface. It can be used to control a wide variety of disc, tape, and solid-state media devices such as DVD/Bluray/CD players, VCRs, and MP3 players. Depending on the supported transfer protocols and data formats, this service may or may not be implemented.

Although most media will be sent across the network as data it may be more efficient to transfer the media data stream using other means. An example is when a personal video recorder is the DMS and a HDTV is the DMR. An Ethernet connection would not be as efficient as an HDMI or component video connection. Using a transfer medium that is not part of the TCP/IP network is called an “out of band” transfer. These transfers are not defined by the UPnP AV specification but are recommended and supported by the manufacturer of the media equipment.

Rendering Control

Most rendering devices contain a number of dynamically configurable attributes that affect how the current content is rendered. For example, video devices, such as HDTVs, allow user control of display characteristics such as brightness and contrast, while audio devices allow control of audio characteristics such as volume, balance, and equalizer settings. The *Rendering Control* service provides control points with the ability to query and adjust any rendering attribute that the device supports.

The *Rendering Control* service enables a control point to:

- Discover the attributes supported by the device.
- Retrieve the current setting of any supported attribute.
- Change the setting of any modifiable attribute.
- Restore the settings defined by a named preset.

DLNA Media Transport

Media Transport defines how content moves across the network. DLNA devices that send or receive any media content via the network must support HTTP 1.1 (including chunked transfer encoding, persistent connections, and pipelining) as the baseline transport mechanism. In addition, Real-time Transport Protocol (RTP) is available as an optional media transport protocol.

Network Stack

The basis for the DLNA *Network Stack* is TCP/IP v4. Every device must implement a DHCP client, and search for a DHCP server when first connected to the network. If a DHCP server is discovered, the device must use the IP address assigned by the server. If no DHCP server is discovered, the device must use Auto-IP to generate a link-local IP address.

Auto-IP uses an implementation dependent algorithm to generate an address in the 169.254/16 range. The first and last 256 addresses in this range are reserved and must not be used. After developing an address, the device must determine if the address is available by using an ARP probe. If the device receives a response, the address is assumed to be in use and the device must generate and test a new IP address.

An Auto-IP configured device must periodically check for the presence of a DHCP server. If a DHCP server is discovered, the device must switch to the IP address allocated to it by the DHCP server. In order to switch between IP addresses, the device must cancel any outstanding UPnP *Discovery* advertisements and re-issue them under the new address.

In addition to IP addressing, UPnP makes extensive use of both the UDP and TCP protocols. *Discovery* is implemented via an HTTP Multicast over UDP. This method is used by devices to advertise their presence to the network and by control points to discover what devices exist on the network. Description, control, and eventing services are delivered via HTTP over TCP.

Network Connectivity

Three network connection technologies are incorporated in the DLNA 1.5 Interoperability Guidelines: 10Base-T and 100Base-T Ethernet (802.3i / 802.3u) for wired connections, WiFi (802.11a / 802.11b / 802.11g) for wireless connections, and Bluetooth for wireless connections for mobile handheld devices such as cell phones and PDAs. Additional network connections such as 1000Base-T Ethernet (802.3ab), WiFi (802.11n) and most Multimedia Over Coax Alliance (MoCA) will be added to the Guidelines in the future. It should also be noted that many other networking technologies such as LonWorks, CeBus, X-10, and Universal Powerline Bus (UPB) could be supported via UPnP Bridges.

Link Protection and Digital Rights Management (DRM)

When commercial content is made available for consumer electronics and mobile devices, it must be protected from unauthorized copying and use. Consumers now expect the capability to store, transfer and use their purchased content on any device at any location connected to wired or wireless digital networks. At this time UPnP and Digital Living Network Alliance (DLNA) are still working to provide a robust and versatile DRM solution capable of interacting with the breadth of proprietary solutions already installed. However, UPnP and DLNA along with many large industry players have approved DTCP-IP and Windows Media Digital Rights Management - Network Device (WMDRM-ND) technologies for the transport of protected content.

Certification

In order to ensure interoperability between DLNA devices, DLNA developed and manages a comprehensive certification program. Vendor products that successfully complete certification are awarded the DLNA Certified accreditation. This lets consumers know the product is fully DLNA compliant and interoperates with other DLNA Certified products. Look for the DLNA Certified logo on products and product packaging.



The initial step in obtaining certification calls for the manufacturer of the product to subject the device to testing utilizing the DLNA's Conformance Test Tool (CTT). The CTT is a suite of tests that are run by the vendor against the product, and validate the devices' compliance with DLNA standards. The test harness for the CTT is a single Windows PC with the device under test connected via a DLNA defined network connection technology (Ethernet, WiFi, Bluetooth). When the device successfully passes the CTT as determined by the CTT's log file, it can begin the formal DLNA certification process.

The formal certification process entails submitting the CTT log and a product's UIC certificate (from UPnP) to DLNA. Next step is scheduling a test session with one of the Independent Certification Vendors (ICV) approved by DLNA. The ICV will test the submitted product per DLNA's Certification Test Plan (CTP) against 3 reference devices of the appropriate device class. For example, the ICV would test a DMP device against 3 DMS reference devices, while a DMC would be tested against 3 DMR and 3 DMS reference devices.

In addition to its formal certification program, DLNA conducts "plugfests" (interoperability workshops) on a regular basis. The plugfests are held each calendar quarter in various locations around the world in order to allow maximum participation from device vendors across the globe. These plugfests provide DLNA member companies the opportunity to test products under development against other member's products using DLNA test tools, and are an excellent dress rehearsal for DLNA certification testing.

CHAPTER 3

RDMS 202 GETTING STARTED

RDMS 202 - Getting Started

Thank you for purchasing the Allegro Reference Digital Media Server (RDMS 202/RDMS 202D). The RDMS 202 is intended as a testing and development tool for OEMs producing DLNA enabled products. The RDMS 202 meets all DLNA 1.5 interface guidelines as a Digital Media Server (DMS). As delivered, the RDMS 202 is provided with DLNA Certification test content for audio, video, and images. The RDMS 202 is available to DLNA Member companies that have signed the DLNA Content Agreement document. The RDMS 202D additionally supports link protection utilizing DTCP-IP and is available to DLNA Member companies that also have a valid DTLA license agreement.

The following manual provides detailed information on setup and use of the RDMS 202 and RDMS 202D in a DLNA 1.5 environment.

RDMS 202 - Getting to Know the RDMS 202



As delivered, the RDMS 202 includes the following items in the box (Figure 3.1):

Figure 3.1 - RDMS 202 box contents

- RDMS 202 - Unit
- RDMS 202 Quickstart Guide (Not Shown)
- Ethernet Cable
- Power Cable
- Allegro PQI USB Memory stick with DLNA test content

Overall, the RDMS 202 is a small, yet powerful device. Measuring only 2.75" x 4.25" x 1.875" and weighing less than 1 pound, the RDMS 202 is a 1.2 GHz ARM based embedded system running embedded Linux along with the RDMS 202 application. Figure 3.2 shows the front of the RDMS 202



Figure 3.2 - RDMS 202 Front Panel

The RDMS 202 has two LED indicators on the front of the device. The green LED (lower LED) comes on once power is applied to the unit. The blue LED (upper LED) is illuminated after power on self test and the processor starts running the RDMS 202 application.

Figure 3.3 shows the top panel of the RDMS 202.

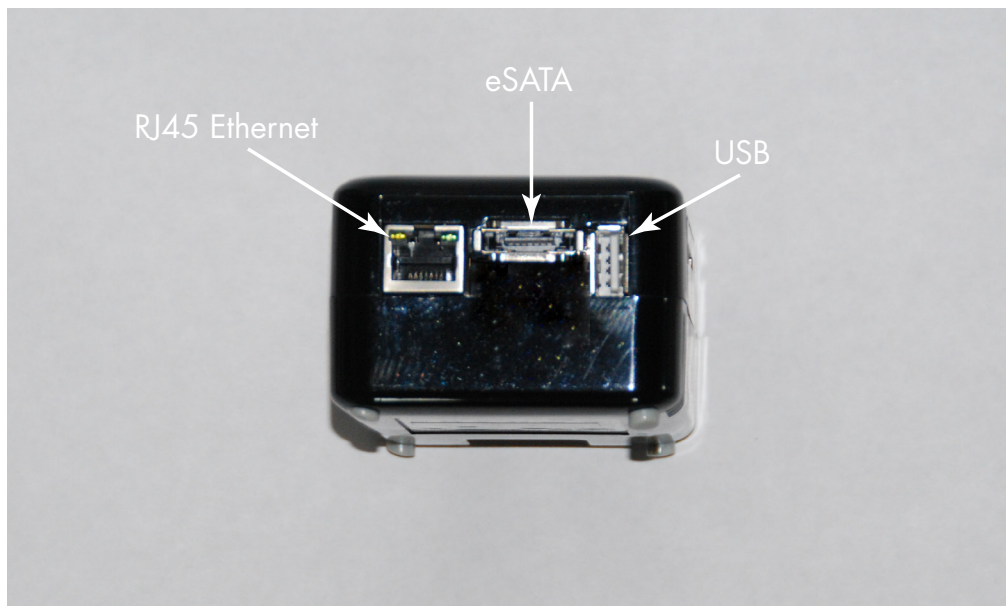


Figure 3.3 - RDMS 202 Top Panel

1. RJ45 Ethernet Connector
2. eSATA - Reserved for future use, do not connect anything to this port
3. USB Connector - This will receive the Allegro PQI USB Memory stick with DLNA Content

Figure 3.4 shows the side panel of the RDMS 202.

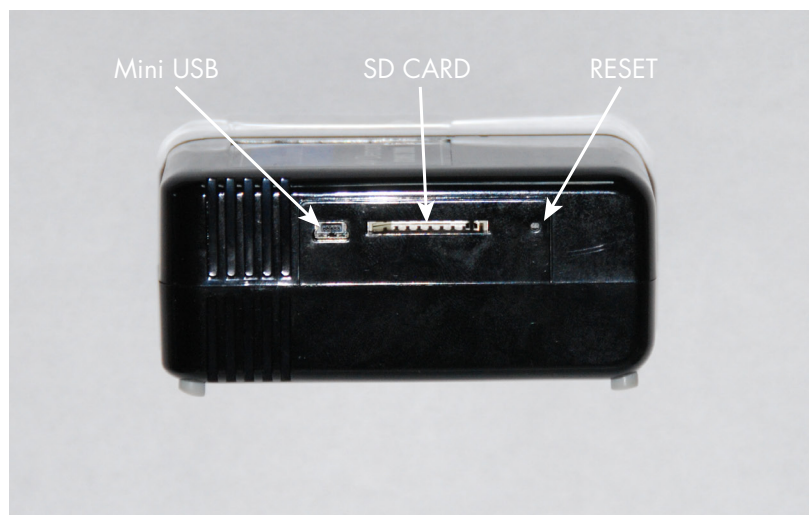


Figure 3.4 - RDMS 202 Side Panel

1. Mini USB - Reserved for future use, do not plug anything into this port
2. SD Card Slot - Reserved for future use, do not plug anything into this port
3. Reset Switch - Can be used to reset RDMS 202 by inserting the end of a paper clip and depressing the reset button

Applicable Documents

The following documents are extremely helpful when working with the RDMS 202 and give insight into the interactions of the RDMS 202 in a DLNA 1.5 environment.

“**DLNA Networked Device Interoperability Guidelines, Volume 1: Architectures and Protocols**”, March 2006, Digital Living Network Alliance, <http://www.dlna.org/industry/certification/guidelines/>

RDMS 202 Requirements

For the RDMS 202 and RDMS 202D to work properly in a DLNA environment, the following minimal requirements must be met:

- A minimum 100-240 V ~ 50/60Hz AC Power
- Ethernet connection to your local network

In addition, to configure preferences or operating parameters, you will need:

- A computer connected to the same local network as the RDMS 202 or RDMS 202D
- The computer must have a browser (Firefox, Netscape, IE, etc.)

Connecting the RDM2 202 or RDMS 202D to a Network

Figure 3.4 illustrates how to connect the RDMS 202 to your local network.

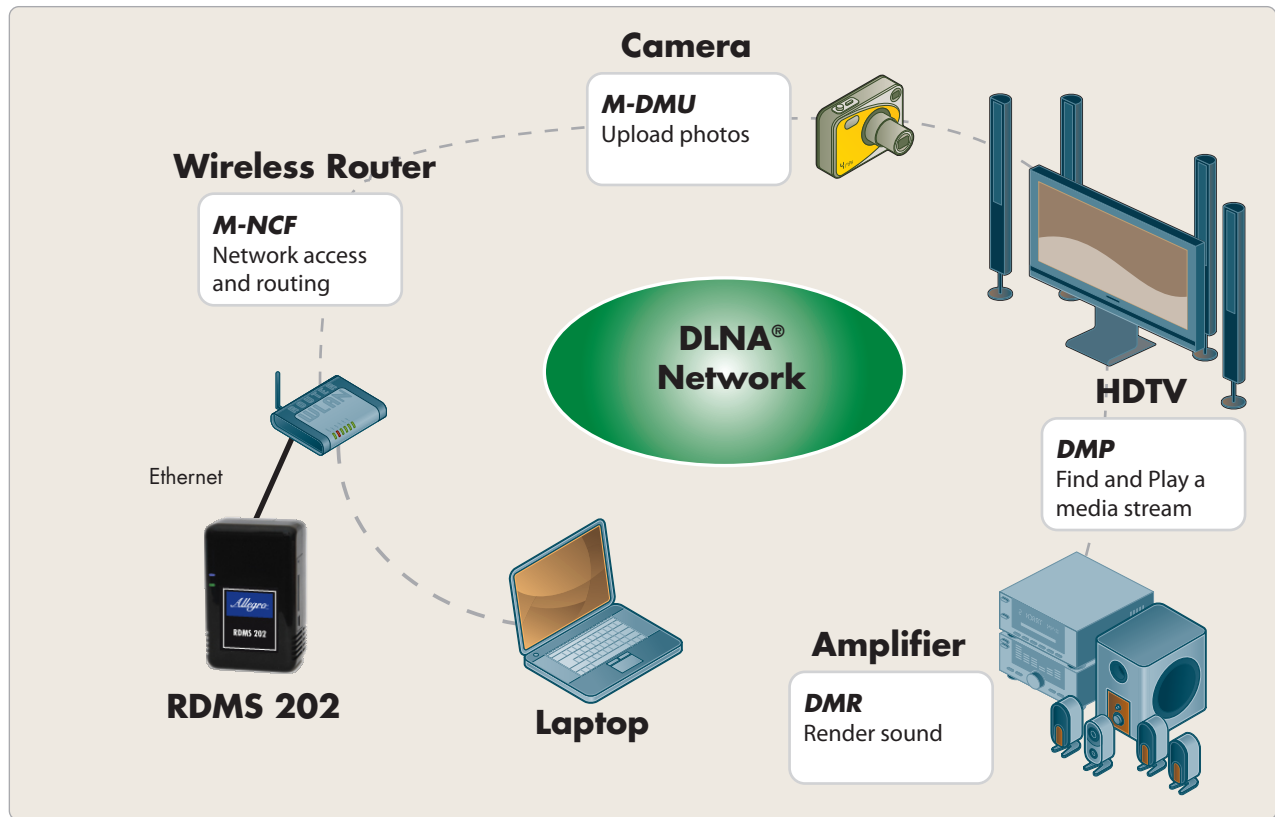


Figure 3.4 - Connecting the RDMS 202 to a Local Network

To connect the RDMS 202 to your local network, use the steps outlined below:

1. Connect one end of an Ethernet cable to the RJ45 connector on the RDMS 202. Connect the other end of the Ethernet cable to a router, switch or hub in your network.
2. Plug the RDMS 202 into a power source.

The green power LED on the front panel will illuminate a steady green when power is applied and RDMS 202 will proceed to boot (less than 10 seconds). Once the RDMS 202 is fully booted, the blue LED will illuminate.

At this point the RDMS 202 is fully powered and ready to serve content to DLNA devices present on your local network. Refer to the following chapter on how to use the RDMS 202 web interface to set preferences and operating parameters for the RDMS 202.

IMPORTANT NOTE: The PQI memory stick shipped with the RDMS 202 or RDMS 202D is pre-loaded with DLNA test content. To prevent accidentally erasing the content, the PQI 32GB USB drive is shipped with write protection enabled. If the RDMS 202 or RDMS 202D will be used to test +UPLOAD+ or if the user plans to add content to the RDMS 202 or RDMS 202D via the built-in web interface, it is advised to make a copy of the PQI contents BEFORE disabling write protection.

CHAPTER 4

RDMS 202 OPERATION

Interacting with the RDMS 202

The RDMS 202 uses a single HTTP interface (Figure 4.1) for setup and configuration of the unit. Using the web interface allows the user to set *Preferences*, check the *Status*, review *Content* parameters and *Add Content* to the RDMS 202.

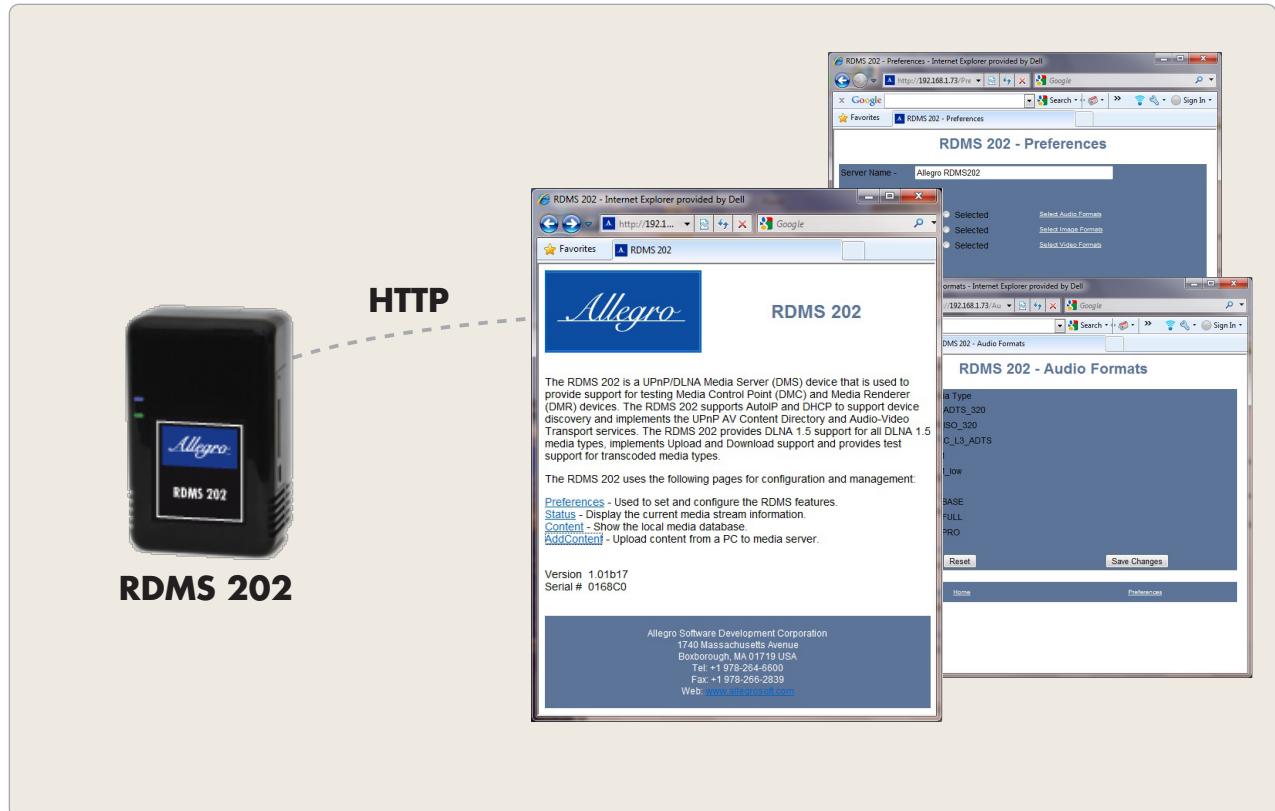


Figure 4.1 - HTTP Interface for RDMS 202

RDMS 202 HTTP Interface

After connecting the RDMS 202 to your local network and turning on the power, the unit will power up and announce itself on the network. If you are using a Windows based PC with UPnP turned on, the RDMS 202 will appear in “My Network Places” (Figure 4.2).

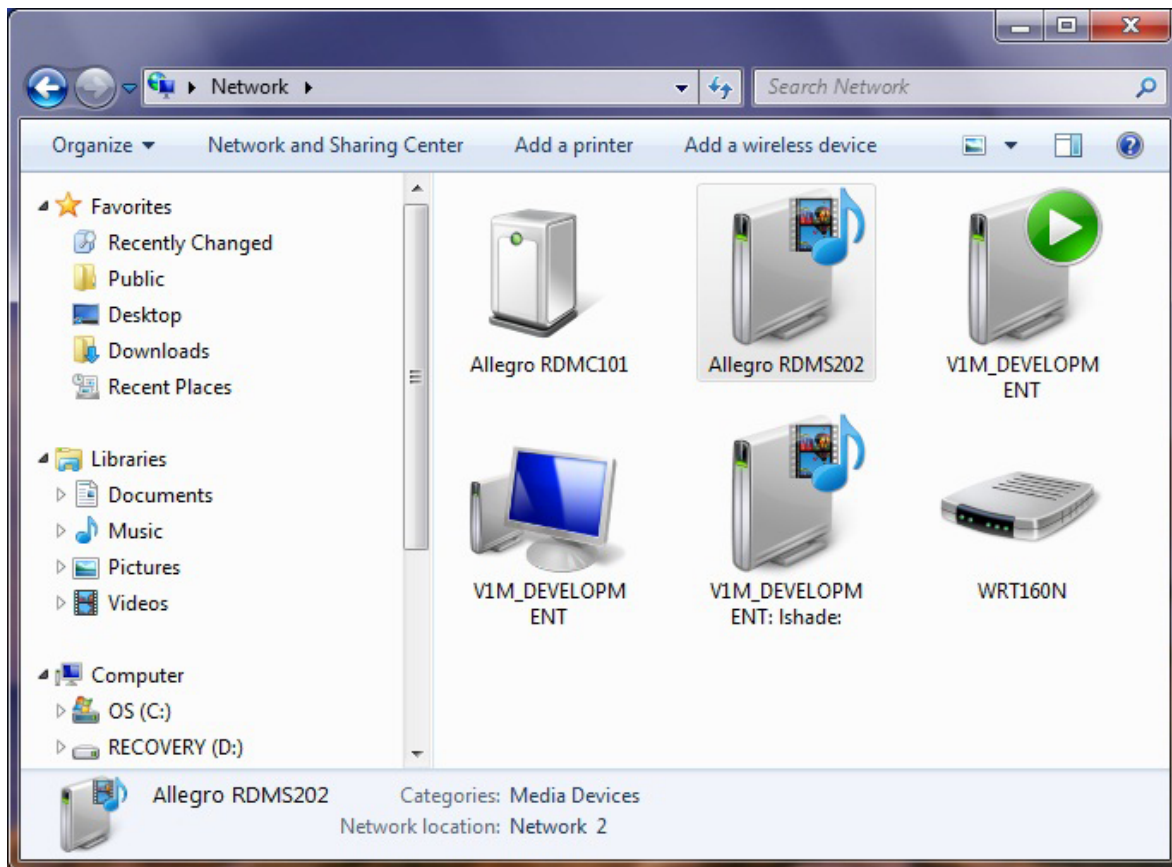


Figure 4.2 - RDMS 202 shown in My Network Places

By double clicking on the RDMS 202 icon, the RDMS 202 UPnP presentation page will appear in your web browser (Figure 4.3).

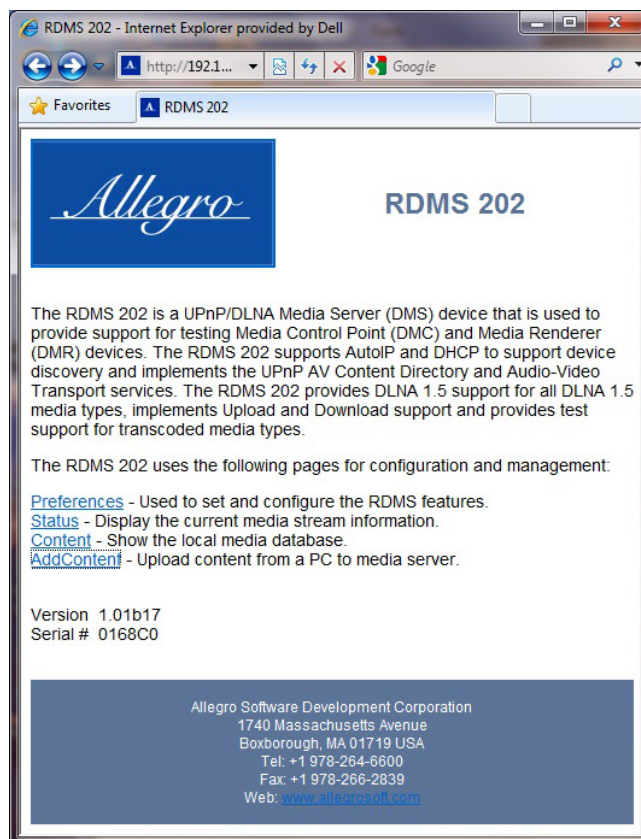


Figure 4.3 - RDMS 202 Presentation Page

From the RDMS 202 Presentation page, users can examine or modify *Preferences*, view the *Status* of the device, browse the media database that represents the *Content*, and users can *Add Content* to the media server. Each capability is further discussed in the following sections.

Preferences

The RDMS 202 *Preferences* (Figure 4.4) allows the user to examine and modify various operating characteristics of the RDMS. Each preference is discussed in detail below.

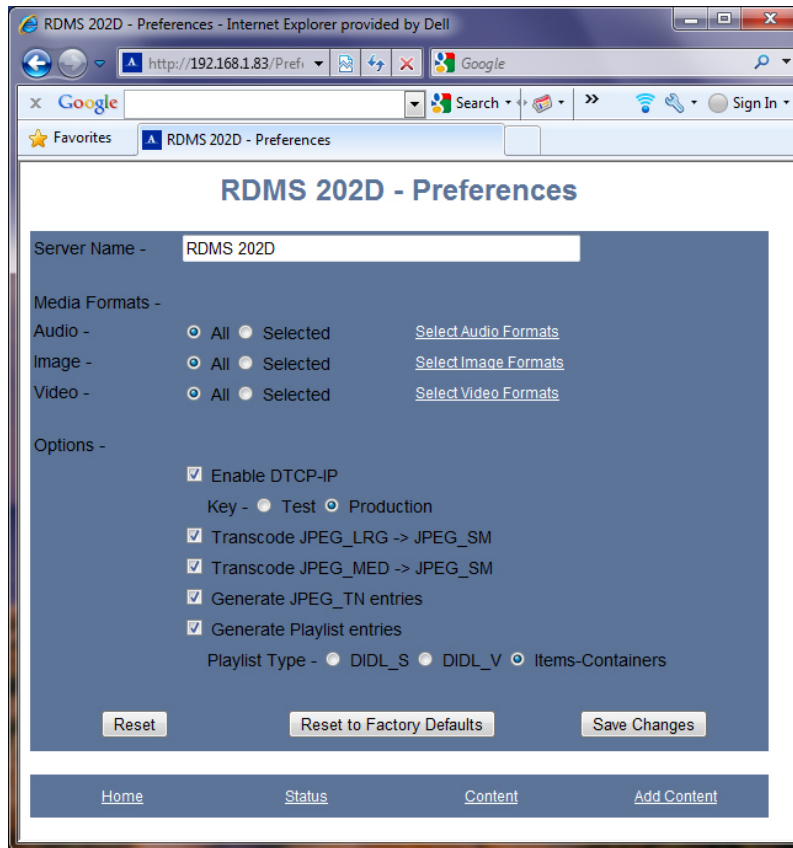


Figure 4.4 - RDMS 202D Preferences Page

Server Name

This allows the user to change the *Server Name* for the RDMS 202. The name entered here will appear as the name of the server when queried on the network. Entries for a *Server Name* can include all printable characters and have a maximum length of 64 characters.

Media Formats

The next section configures what media formats the RDMS 202 will publish as available content. To configure which formats are published, click on the *Selected* radio button and then click on the associated link to choose which formats will be published. Different formats can be selected for *Audio* (Figure 4.5), *Image* (Figure 4.6) and *Video* (Figure 4.7). When choosing which formats will be published, check (or uncheck) the box next to the associated media type and choose *Save Changes* at the bottom of the screen. Pressing *Reset* will return the selected media formats to their last known state.

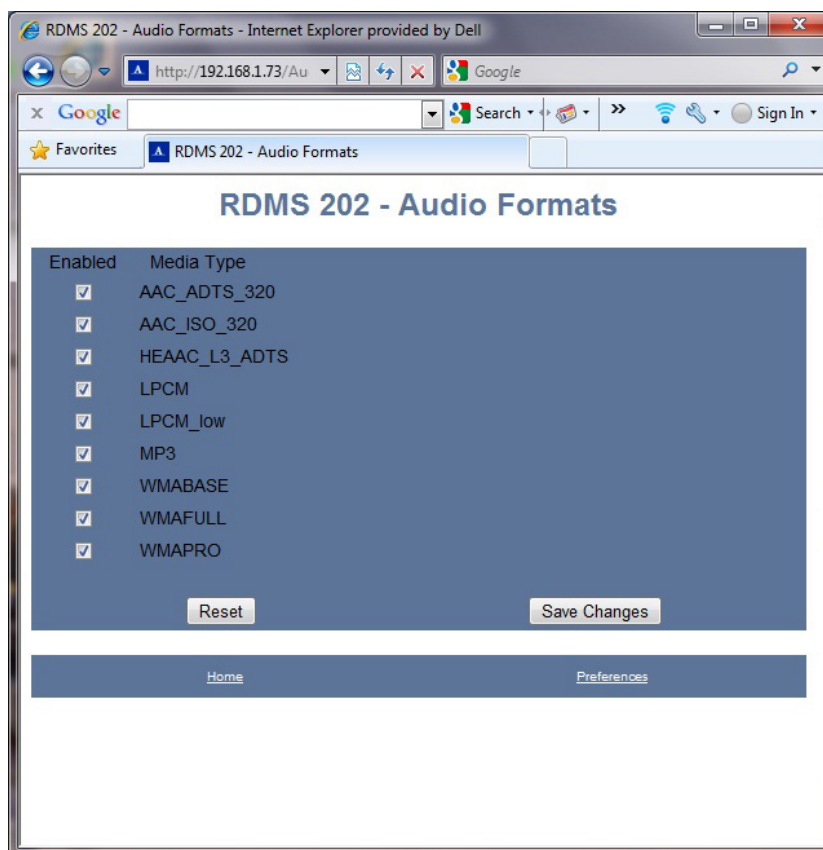


Figure 4.5 - RDMS 202 Audio Formats Page

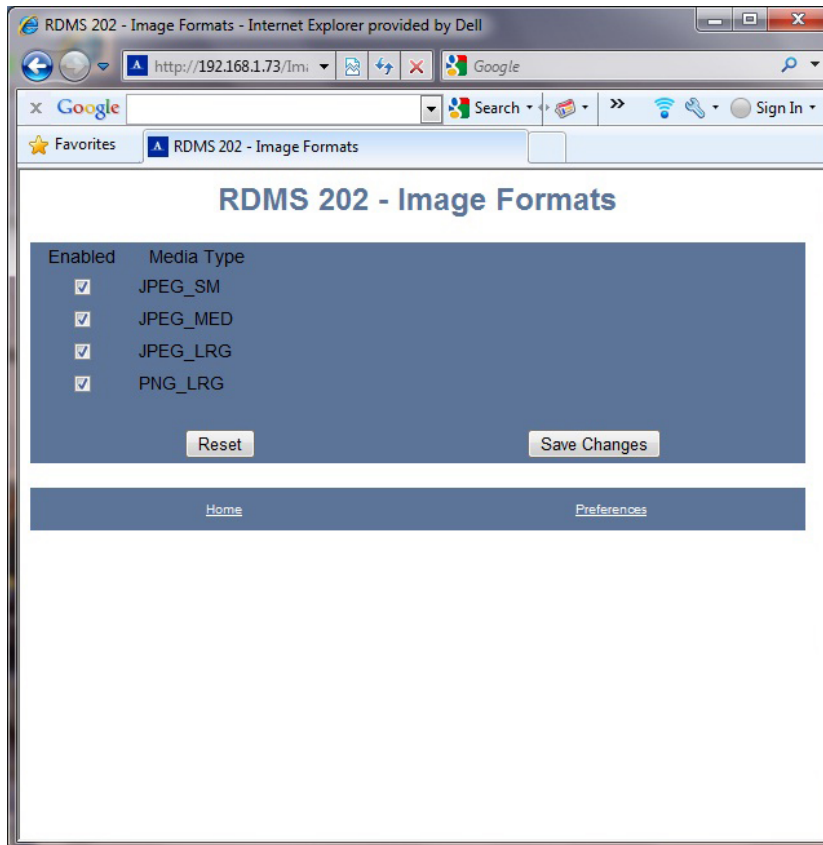


Figure 4.6 - RDMS 202 Image Formats Page

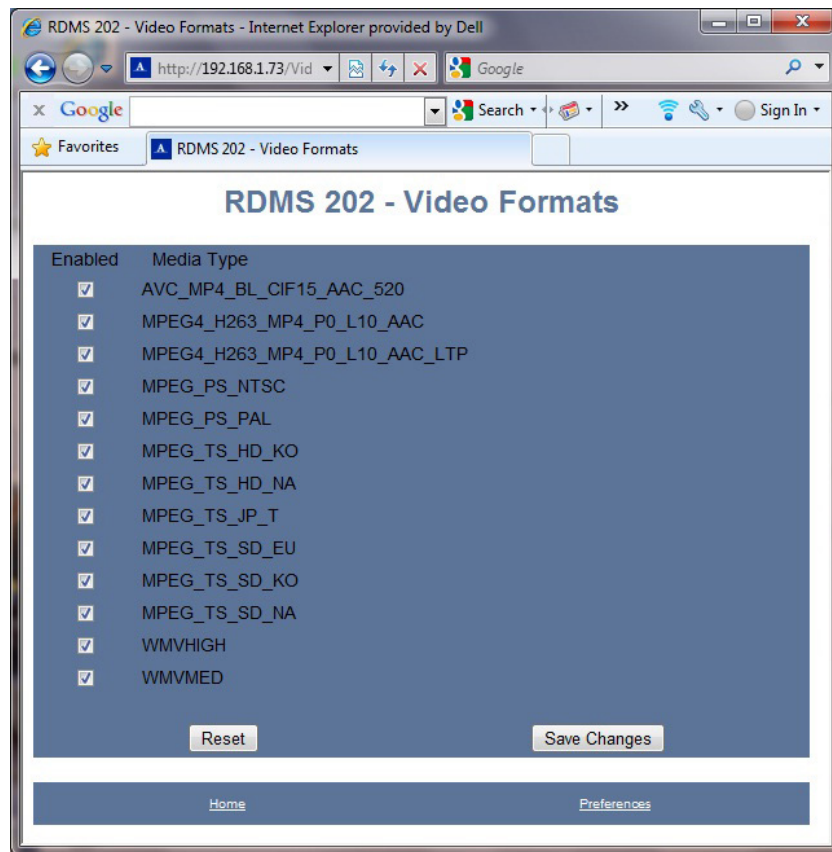


Figure 4.7 - RDMS 202 Video Formats Page

There are additional DTCP-IP, Transcoding, Thumbnail, and Playlist *Options* available for the RDMS 202 and RDMS 202D (See Figure 4.8).

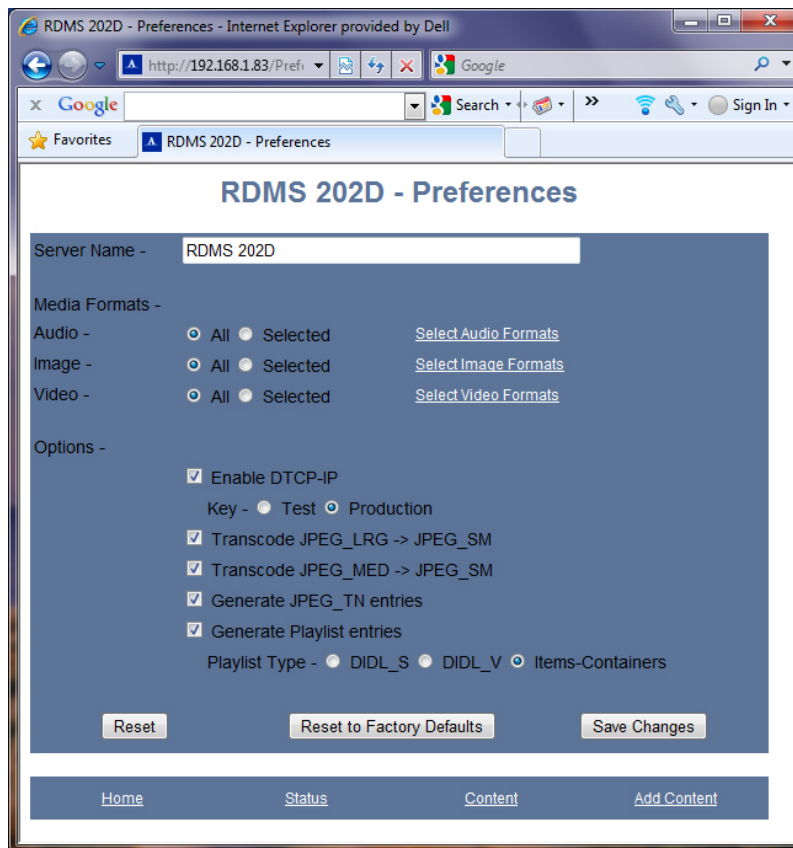


Figure 4.8 - RDMS 202D Preferences Page

RDMS 202D Only

The RDMS 202D supports DTCP-IP for link protection. If DTCP-IP is needed, click the radio button next to *Enable DTCP-IP*. The RDMS 202D supports 2 different sets of keys when employing DTCP-IP. Often, during development and initial testing, DTCP-IP *Test* keys are used to verify operation of DTCP-IP. Later the RDMS 202D can utilize *Production* DTCP-IP keys for detailed component testing using DTCP-IP.

RDMS 202 and RDMS 202D

Realtime Transcoding of JPEG images

There are two options to support realtime transcoding of large and medium JPEG images to small. If this support is required, click the radio button next to the type of transcoding desired.

Thumbnail entries for JPEG images

The RDMS 202 is capable of automatically generating thumbnail images. To enable this functionality, click the radio button next to *Generate JPEG_TN entries*.

Playlists

The RDMS 202 can generate *Playlists* in three different formats (*DIDL_S*, *DIDL_V*, and *Container*) with item entries. *DIDL_S* and *DIDL_V* are individual DLNA format files that contain a list of entries that make up a playlist. The *Container* option represents a playlist such as All Video items as another container/folder in the directory structure. To enable this functionality click the radio button next to *Generate Playlist entries* and then select one of the three formats available.

Reset

The *Reset* button returns all setting to their last known values.

Reset to Factory Defaults

The *Reset to Factory Defaults* will return all preferences and settings to the original factory defaults.

Save Changes

After making changes to any preferences or settings it is important to click the *Save Changes* button for the RDMS 202 to accept new parameters and reconfigure accordingly. In some cases the RDMS 202 will automatically reboot to properly implement the new preferences settings.

Status

The RDMS 202 *Status* page (Figure 4.9) summarizes the overall status of the RDMS 202 including the number of Audio, Image, and Video items available on the server. Additionally, users can see if the RDMS 202 is currently serving files to a renderer or media player.

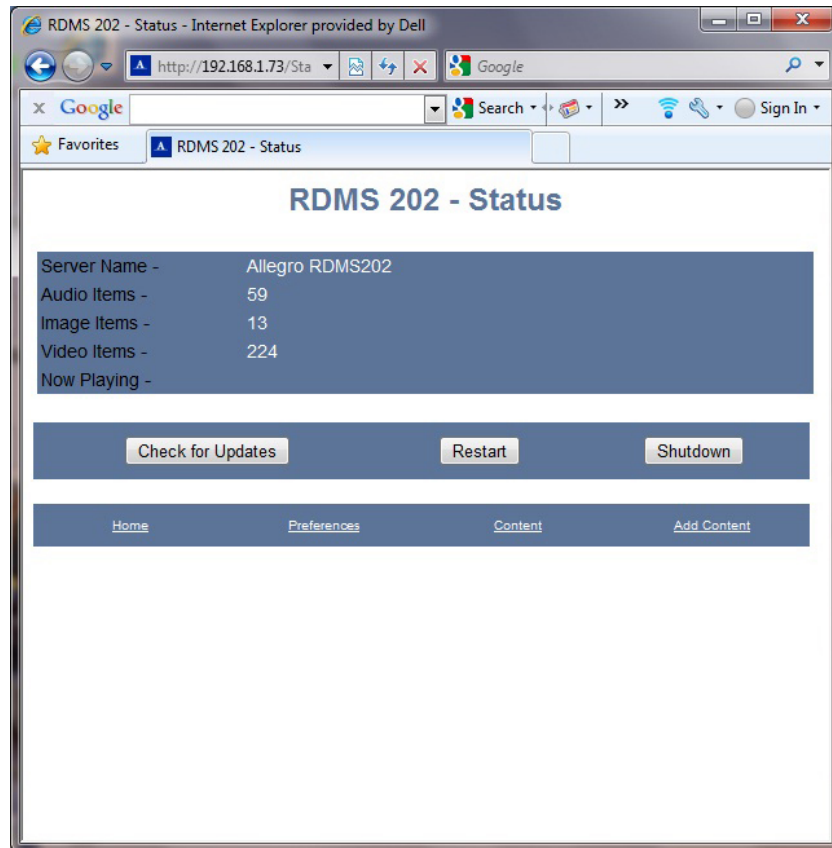


Figure 4.9 - RDMS 202 Status Page

There are three additional operations that can be performed from this page.

Check for Updates

The RDMS 202 and RDMS 202D have the ability to remotely pull firmware updates from the Allegro Digital Firmware Server (ADFS). Once a user has clicked on the *Check for Updates* button the RDMS 202 or RDMS 202D contacts the ADFS and displays the current firmware release number along with a list of other firmware updates available (Figure 4.10). Select the appropriate release and click on the *Install* icon. The RDMS 202 will proceed to download the revised firmware, store it internally and reboot.

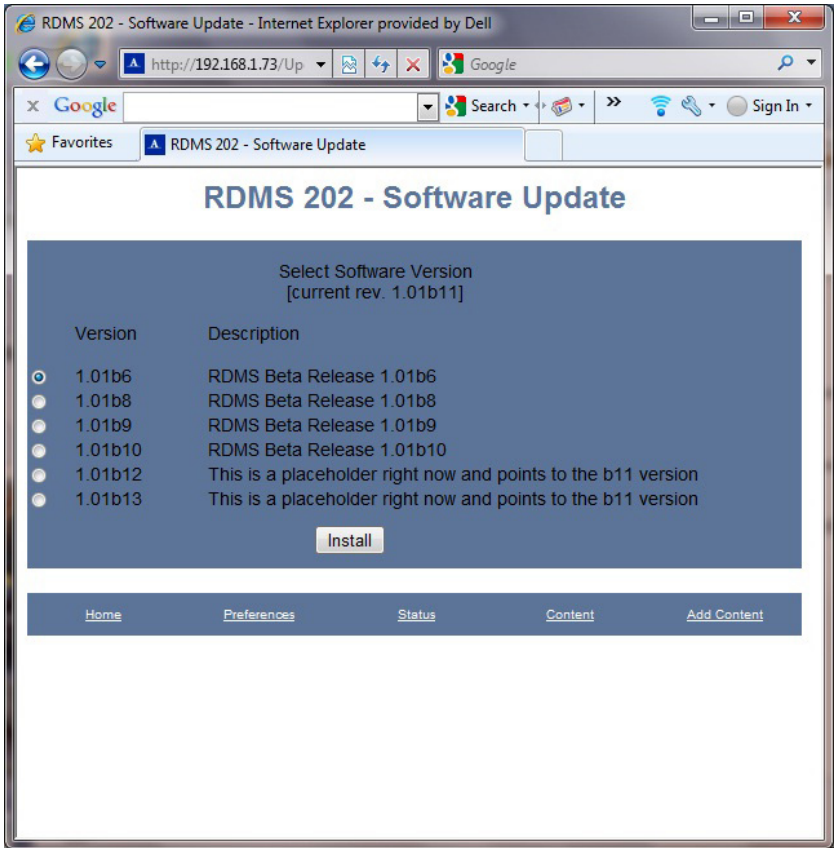


Figure 4.10 - RDMS 202 Software Update Page

Restart

This button causes the RDMS 202 or RDMS 202D to shutdown gracefully and restart. After clicking on this button the RDMS will immediately stop serving any files and restart.

Shutdown

After clicking on the *Shutdown* button the RDMS 202 or RDMS 202D will immediately stop serving any files and perform a graceful shutdown of the server application. Power is not under software control so the RDMS 202 will remain powered but will not serve media files until power is cycled.

Content

The RDMS 202 *Content* page (Figure 4.11) allows a user to traverse the structure of the local content database. There are six commands located in the upper left that are used to traverse the database structure (*Top*, *Up*, *Previous Page*, *Next Page*, *Previous Item* and *Next Item*).

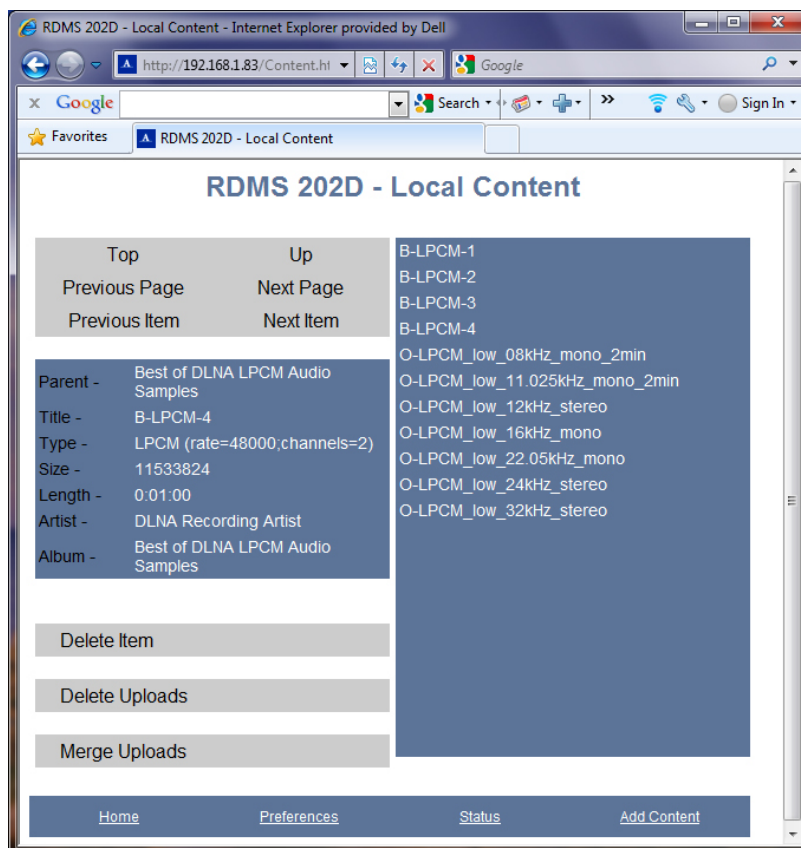


Figure 4.11 - RDMS 202 Local Content Page

Top

Clicking on *Top* displays the highest level of the database structure.

Up

Clicking on *Up* changes the display to represent the next highest level in the structure of the database.

Previous Page

When displaying items within a container, if the container has more items than will fit on a single page and the display is currently showing items from the second or higher pages, clicking *Previous Page* will display items from the container immediately preceding the first item on the current page.

Next Page

When displaying items within a container and there are more items than can be displayed on a single page, clicking *Next Page* will display items from the container immediately following the last item on the current page.

Previous Item

Once an item is selected in the blue area on the right side of the screen, information regarding that item is displayed in the blue area on the left side of the screen. Clicking *Previous Item* will display the item immediately before the currently selected item. If the previous item is on the previous page the display will refresh with a list of previous items.

Next Item

Once an item is selected in the blue area on the right side of the screen, information regarding that item is displayed in the blue area on the left side of the screen. Clicking *Next Item* will display the item immediate following the currently selected item. If the next item is on the next page the display will refresh with a list of items following the current item.

Delete Item

Once an item is selected in the blue area on the right side of the screen, the user can delete the item by clicking on the *Delete Item* icon. The highlighted item will be deleted from the RDMS.

Delete Uploads

The RDMS 202 and RDMS 202D units support the DLNA Upload option which allows a DLNA device with a +UP+ controller option to send content over the network to the RDMS. When testing the DLNA Upload option, it is often useful to return the test media server to a known state. By clicking on the *Delete Uploads* command, all content uploaded from a DLNA device with a +UP+ controller option is removed from the RDMS.

Merge Uploads

If the *Merge Uploads* command is selected, the RDMS will use the metadata originally attached to the previously uploaded content to merge the uploaded content into the current RDMS content directory.

Add Content

The RDMS 202 *Add Content* page (Figure 4.12) allows a user to supplement existing content on the RDMS. Users are given the ability to enter specific metadata such as Title, Artist, Album and Genre. Users are then asked to provide the type of media class (Audio, Image or Video) and select the appropriate Media Type. The pull down list shows all formats that are available. The user is then must click on the Browse icon to select the file to upload to the RDMS. Once the file has been selected click on the Add Content icon to upload the new content to the RDMS.

The screenshot shows a web browser window titled "RDMS 202D - Add Content - Internet Explorer provided by Dell". The address bar shows "http://192.168.1.67/". The page has a blue header with the title "RDMS 202D - Add Content". Below the header, there is a form with the following fields and controls:

- Title -
- Artist -
- Album -
- Genre -
- Media Class - ☒ Audio ☐ Image ☐ Video
- Media Type -
- PC File -
-

At the bottom of the form, there is a navigation bar with four links: [Home](#), [Preferences](#), [Status](#), and [Content](#).

Figure 4.12 - RDMS 202 Add Content Page

CHAPTER 5

MEDIA FORMATS, CDS AND USB DRIVE

Understanding the RDMS Content Directory Service

Media formats play an important role to enable DLNA interoperability between media servers, media renderers and media players from multiple vendors. The DLNA defined media types that a media server, media renderer or media player support dictate which content can be shared and in some cases if link protection is also required. Unlike typical PC based applications, knowing filenames and locations does not provide enough data about content to create a DLNA enabled server. Filenames and locations must be augmented with additional metadata that describe media type, protection requirements, album art, size, duration in addition various other metadata. The following code fragment is from `MsAppCds.c` and highlights the data typically stored for a single Content Directory Service (CDS) entry.

```
/*
    The sample CDS is built around a data table made from an array of
    cdsAppObject elements. Each element corresponds to an object in the CDS.
*/
typedef struct {
    Unsigned32                fType;
    MsCdsObjectId             fId;
    MsCdsObjectId             fParentId;
    char                      fTitle[kMsMaxObjectTitleLength];
    char                      fCreator[kMsMaxPropertyStringLength];
    char                      fDate[kMsMaxPropertyStringLength];
    cdsPersonaProperty        fPlayer;
    cdsAlbumArtUriProperty    fAlbumArtUri;
    Unsigned32                fSize;
    Unsigned32                fDuration; /* in milliseconds */
    MsCdsObjectId             fRefId;
    Boolean                   fRestricted;
    cdsObjectClass            fClass;
#ifdef RuDlnaCompatible
    avMediaFormat             fProfile;
#endif
#ifdef RomDtcpipSource
    dtConstraint              fCopyConstraint;
#endif
    char                      fResource[32];
    char                      fMimeType[32];
    Boolean                   fSearchable;
    cdsClassAction            fActionClass[kMsMaxClassActions];
} cdsAppObject, *cdsAppObjectPtr;
```

For a DLNA server to function properly, the metadata stored in the database describing the content must accurately represent the media content itself (names, locations, media types, etc.). Figure 5.1 is a block diagram representation of the RDMS 202 and RDMS 202D. Content for the RDMS 202 is stored on the external 32 gigabyte (GB) PQI USB drive. The CDS with the required metadata is stored internal to the unit.

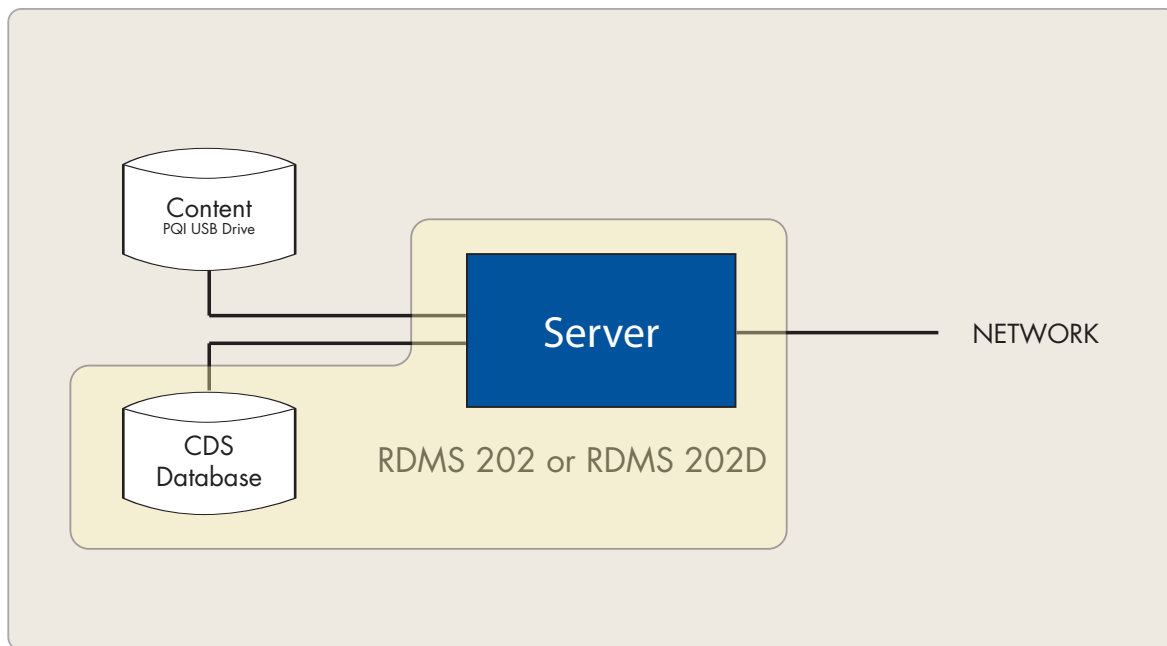


Figure 5.1 - RDMS 202 and RDMS 202D Block Diagram

There are various ways to add and delete content from the RDMS 202 or RDMS 202D. Additionally, a method has been included to rapidly return the RDMS 202 or RDMS 202D to a known operational state. This is specifically useful when testing *Upload* (+UP+) capabilities. Although the commands to accomplish each of these tasks have been presented previously, the following sections provide additional detail regarding the underlying architecture and operational characteristics.

PQI Directory Structure

The 32GB PQI USB drive provides storage for all media content on the RDMS 202 and RDMS 202D. The corresponding CDS and supporting metadata however, is NOT stored on the drive and stored using internal memory. The PQI drive is organized with two (2) main directories off the root structure: *DLNA Content* and *Upload* as shown in Figure 5.2. The *DLNA Content* directory, as delivered from the factory, contains the latest DLNA Certification content. The *Upload* directory will contain all files that have been uploaded to the RDMS 202 or RDMS 202D using the *Upload* (+UP+) feature in addition to content added directly to the RDMS 202 or RDMS 202D using the *Add Content* web page.

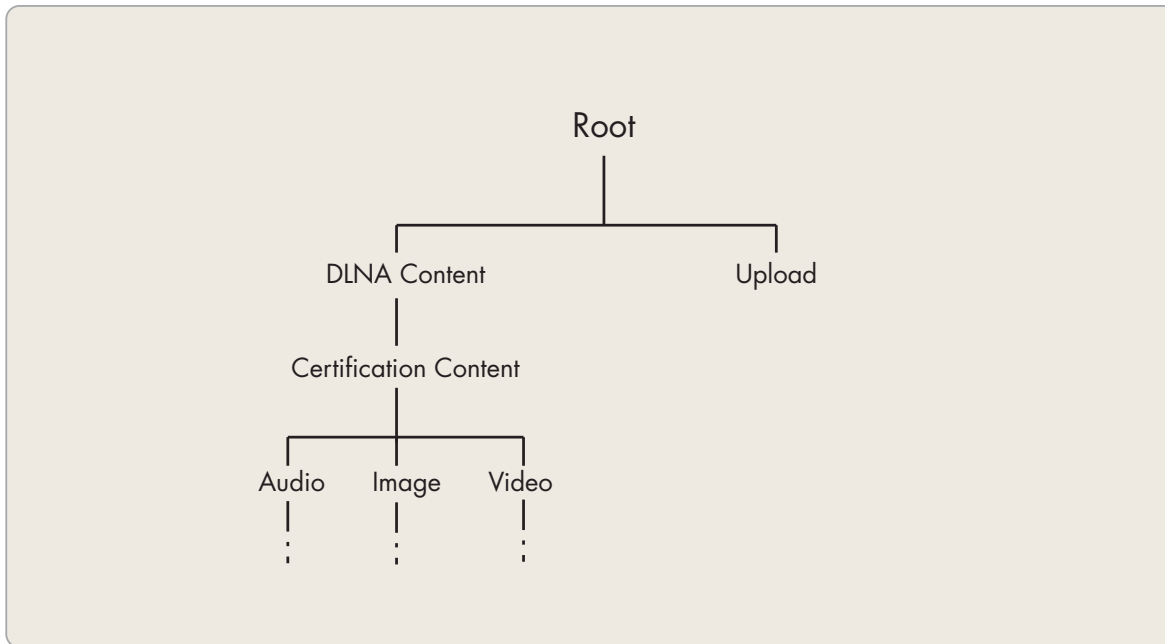


Figure 5.2 - PQI USB Drive Directory Structure

WARNING:

The PQI USB drive is delivered pre-populated with DLNA Certification content. To protect against inadvertently corrupting the pre-populated data, the PQI USB drive is shipped with the write protection switch enabled. Allegro highly recommends making a backup copy of the contents of the PQI USB drive BEFORE disabling write protection on the drive. Figure 5.3 shows PQI USB drive write protection switch in the UNLOCKED position. The PQI USB drive must be in the UNLOCKED position before content can be added or deleted from the RDMS 202 or RDMS 202D.



Figure 5.3 - PQI USB Drive Write Protect Switch

Reset to Factory Defaults

When developing or testing DLNA enabled devices it is often useful to have the ability to return the RDMS 202 or RDMS 202D to a known state after testing *Upload* (+UP+) capability. To quickly bring the RDMS 202 or RDMS 202D back to a known state, press the “*Reset to Factory Defaults*” button on the RDMS 202 or RDMS 202D *Preferences* page. Once the button is pressed, the RDMS 202 and RDMS 202D firmware re-initializes the internal CDS database with pre-defined metadata assuming all DLNA Certification content is available on the external PQI USB drive. It is important to note, even if content previously uploaded to the RDMS 202 or RDMS 202D is present on the PQI USB drive it will not be recognized. The RDMS 202 and RDMS 202D assume that only the delivered DLNA Certification content is available on the PQI USB drive immediately after being *Reset to Factory Defaults*.

Adding Content to RDMS 202 or RDMS 202D

There are two (2) methods for augmenting existing content on the RDMS 202 or RDMS 202D as shown in Figure 5.4. The first method exercises the *Upload* (+UP+) capabilities built into the RDMS 202 or RDMS 202D. A Media Controller such as the RDMC 101, is used to browse and select source content from a server and initiate the transfer from the source server to the RDMS 202 or RDMS 202D. In this case, the existing metadata on the source server is utilized to update the CDS on the RDMS 202 or RDMS 202D. The RDMS 202 or RDMS 202D stores the metadata in the internal CDS and then deposits the incoming content in the *Upload* directory on the PQI USB drive. The second method makes use of the *Add Content* feature available by communicating directly with the RDMS 202 or RDMS 202D via a web browser. This method allows a user to add content and populate all relevant metadata fields stored in the CDS. Additionally, it is important to note there is no media type checking performed by the RDMS 202 or RDMS 202D for uploaded content. As many media types don't store metadata about the file internally, the RDMS 202 and RDMS 202D use the provided metadata to categorize the content.

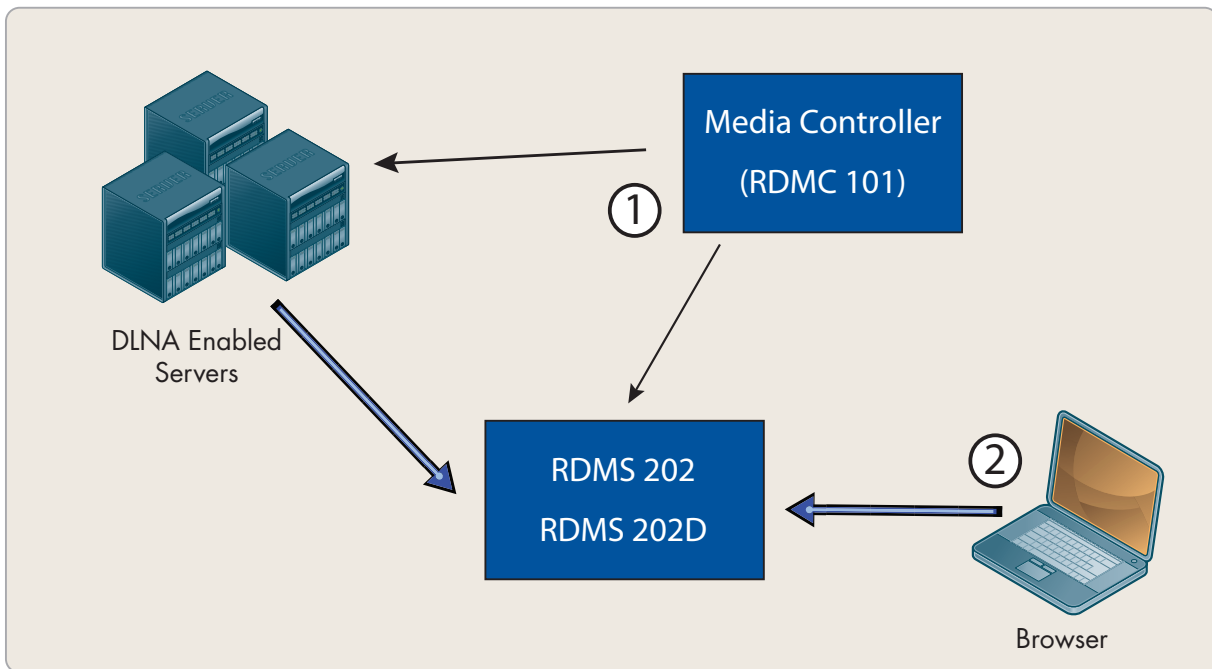


Figure 5.4 - Methods for Adding Content to RDMS 202 or RDMS 202D

IMPORTANT NOTE: The PQI USB drive write protect switch must be in the UNLOCKED position before content can be added to the RDMS 202 or RDMS 202D

Merge Upload

[NOTE – THIS IS NOT HOW THE UNIT ACTUALLY WORKS WITH VERSION 1.01b11
– THERE COULD BE VARIOUS BUGS HERE – WHAT IS SHOWN BELOW REPRESENTS MY
UNDERSTANDING OF WHAT WAS INTENDED]

After content is added to the RDMS 202 or RDMS 202D, when queried the new content is presented as part of the CDS entries for the *Upload* container. For ease of browsing, the RDMS 202 and RDMS 202D offer the ability to merge the uploaded content with the existing *DLNA Certification* content based on the CDS metadata for each item in the *Upload* container. Once the *Merge Uploads* button is selected, the CDS entries are changed to reflect the associated container within the *DLNA Certification* content and the content is also moved from the *Upload* directory on the PQI USB drive to the associated directory within the *DLNA Certification* directory structure. Once this step is performed, the uploaded content is considered to be part of the *DLNA Certification* data and will not be deleted or removed by the *Delete Uploads* function.

Deleting Content from the RDMS 202 or RDMS 202D

The RDMS 202 or RDMS 202D supports two (2) methods for deleting content. The first method deletes a single item while the second deletes all uploaded content not yet merged with the *DLNA Certification* content.

Deleting a single item

Deleting a single content item on the RDMS 202 and RDMS 202D is easily performed. Using the *Content* page served from the RDMS 202 or RDMS 202D, navigate and select the content to be deleted. Once selected, click on the *Delete Item* button and the RDMS 202 or RDMS 202D will remove the content from the CDS as well as erase the file from the PQI USB drive. There are no restrictions for deleting content from the RDMS 202 or RDMS 202D. Typically, this functionality is used to delete a recently uploaded piece of content without deleting all uploaded files. It is important to note that if an item is deleted from the *DLNA Certification* content directories and the RDMS 202 or RDMS 202D is returned to factory defaults by clicking on the *Reset to Factory Defaults* icon, the internal CDS could indicate a file is available when it is not present on the PQI USB drive. Additionally, files may be present on the USB drive but not represented in the CDS.

IMPORTANT NOTE: The PQI USB drive write protect switch must be in the UNLOCKED position before content can be removed from the RDMS 202 or RDMS 202D

Delete Uploads

[NOTE – THIS IS NOT HOW THE UNIT ACTUALLY WORKS WITH VERSION 1.01b11
– THERE COULD BE VARIOUS BUGS HERE – WHAT IS SHOWN BELOW REPRESENTS MY
UNDERSTANDING OF WHAT WAS INTENDED]

The *Delete Uploads* function is a quick and easy method for deleting all uploaded content that has not been merged into the *DLNA Certification* content. All content that is presented as part of the *Upload* container is deleted from the PQI USB drive and the corresponding CDS entries are removed.

IMPORTANT NOTE: The PQI USB drive write protect switch must be in the UNLOCKED position before content can be removed from the RDMS 202 or RDMS 202D