RadioVIS Technical Specification

RVIS01 V1.0.0 (2009-09)

An application to enhance broadcast audio services with IP-delivered visuals.

The RadioDNS Project

http://radiodns.org/
feedback@radiodns.org

Important notice

Individual copies of the present document can be downloaded from http://radiodns.org/docs

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF).

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other RadioDNS documents is available at http://radiodns.org/docs

If you find errors in the present document, please send your comment to the document editors feedback@radiodns.org

Copyright Notification

Contents

Intellectual Property Rights

The contributors to the present document have declared that they have no IPRs in respect of their contributions. However no investigation, including IPR searches, has been carried out, so no guarantee can be given as to the existence of other IPRs which are, or may be, or may become, essential to the present document.

Foreword

Radio functionality is often included in devices with colour displays capable of showing texts and images.

An existing specification allows the transmission of slideshow over DAB Digital Radio [1]. The present document defines a similar methodology based on IP which, in conjunction with RadioDNS [2], allows the transmission of slideshow images and text to support audio services carried over multiple audio delivery protocols such as VHF/FM and IP.

1. Scope

The present document defines the protocol for RadioVIS to allow implementation from both a service provider and client perspective.

2. References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] TS 101 499: 'SlideShow; User application specification'

[2] RDNS01: 'RadioDNS Technical Specification'

[3] RFC 1035 (1987): 'Domain Names – Implementation and Specification'

[4] RFC 2782 (2000): 'A DNS RR for specifying the location of services (DNS SRV)'

[5] 'Stomp Protocol Specification, Version 1.0', http://docs.codehaus.org/display/STOMP/Protocol

[6] RFC 793 (1981): 'Transmission Control Protocol'

[7] IEC 62106 (2009): 'Specification of the Radio Data System (RDS) for VHF/FM sound broadcasting in the frequency range from 87,5 MHz to 108,0 MHz'

[8] ISO 3166-1, 'Codes for the representation of names of countries and their subdivisions – Part 1: Country codes'

[9] ETSI EN 300 401: 'Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers'

[10] ETSI ES 201 980, 'Digital Radio Mondiale (DRM); System Specification'

[11] ETSI TS 102 386, 'Digital Radio Mondiale (DRM); AM signalling system (AMSS)'

[12] NRSC-5-B:2008, 'In-band/on-channel Digital Radio Broadcasting Standard'

[13] RFC 2616 (1999): 'Hypertext Transfer Protocol – HTTP/1.1'

[14] ISO 8601:2004 (2004): 'International standard date and time notation'

[15] RFC 1738 (1994): 'Uniform Resource Locators (URL)'

[16] T.81 (1993): 'Information Technology – Digital Compression and Coding of Continuous-Tone Still Images – Requirements and Guidelines'
[17] ISO/IEC 15948:2004: 'Portable Network Graphics (PNG): Functional Specification'

[18] Annex A TS 101 499 V2.2.1 (2008-07): 'APNG 1.0 Specification – Animated Portable Network Graphics'

3. Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

RadioDNS: centralised lookup for radio services, allowing the resolution of broadcast parameters to an authoritative FQDN as detailed in RNDS01 [2]

service: an audio service such as a talk or music radio station

service provider: the organisation providing a service

slide: a single image containing informative content related in someway to the service audio

slideshow: the presentation of slides controlled by the service provider

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AMSS	Amplitude Modulation Signalling System
APNG	Animated Portable Network Graphics
CNAME	DNS Canonical Name record
DAB	Digital Audio Broadcasting
DLS	Dynamic Label Segment
DNS	Domain Name System
DRM	Digital Radio Mondiale
FQDN	Fully Qualified Domain Name
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
JPEG	Joint Photographic Experts Group
PNG	Portable Network Graphics
RDS	Radio Data System
SRV	DNS nameserver Service record
Stomp	Streaming Text Orientated Messaging Protocol
TCP	Transmission Control Protocol
URL	Uniform Resource Locator
VHF/FM	Very High Frequency/Frequency Modulation

4. Application discovery and transport selection

4.1 Application discovery

A device must be capable of resolving the authoritative FQDN for an audio service via the methodology defined in the RadioDNS specification [2].

A further DNS lookup [3] must then be performed by pre-pending the RadioVIS application SRV record prefix to the authoritative FQDN and performing an SRV record lookup. In this instance, the prefix is:

```
_radiovis._tcp
```

If an SRV record is successfully resolved, this audio service supports the RadioVIS application. The record also provides the host(s) and port(s) where the relevant RadioVIS application server resides:

```
_radiovis._tcp.rdns.foldermedia.com  service = 0 100 61613 174.129.164.232.
```

Be aware that more than one result can be returned for a single lookup. See the SRV Record specification [4] for a detailed explanation on handling SRV resolution results.

4.2 Transport selection

The present document defines two transport methods, Stomp and Comet.

The SRV Record Port Number entry defines the server port that the client should connect to for the Stomp transport. If the Stomp transport is not supported by the service provider the Port Number specified in the SRV record must be 0 (zero).

For service providers implementing a RadioVIS server:

- You are **STRONGLY RECOMMENDED** to implement Stomp on port 61613 due to the possibility that Stomp traffic on a non-standard port will be rejected by firewall/proxy configurations
- You **MAY** implement Stomp on any other port
- You **MAY** implement Comet and if implemented it **MUST** be on port 80
- You **MUST** implement at least one of the protocols
- You **MAY** implement both protocols
- You **MAY** implement Stomp on more than one port

For manufacturers and developers implementing a RadioVIS client:

- You are **STRONGLY RECOMMENDED** to implement the Stomp protocol
- You **MAY** implement the Comet protocol
- You **MUST** use the Stomp protocol in preference to the Comet protocol
- You **MUST** implement at least one of the protocols

You should recognise that Stomp will not traverse strict HTTP proxies. Comet will correctly pass through most HTTP proxies. However, Stomp is always recommended over Comet due to its more efficient nature as a true push notification transport.

5. Stomp transport

The Stomp implementation is based upon the Stomp specification [5]. A client connects to the server and then subscribes to specific channels or 'topics'. Once subscribed, text-based messages are then received related to the chosen topics.

Stomp messages are sent and received within a frame. All message bodies should be encoded as UTF-8 with Unicode character encoding. All frames are terminated using a NULL ASCII character `^@` (control-@).

The present document defines the essential frames for RadioVIS below. Potentially the server could also send other frames, including `ERROR`. It is recommended that a device does not surface any errors to the user but retain previously received text or visual content.

Please refer to the Stomp specification [5] for specific and detailed protocol definitions.

5.1 Connecting to the server

Connect to the derived destination on a TCP [6] socket and send a CONNECT frame to the server:

```
CONNECT

^@
```

The server will respond with an acknowledgement frame:

```
CONNECTED
session: <session-id>

^@
```

5.2 Subscribing to a topic

Send a SUBSCRIBE frame to the server, indicating the topic to subscribe to:

```
SUBSCRIBE
destination: <topic-id>
ack: auto

^@
```

This will result in messages received by a client being deemed by the server to have been acknowledged. This means the client will not need to send an acknowledgement back.

The construction of the `topic-id` is detailed in the next section of the present document.

5.3 Topic ID construction

The topic ID used to subscribe is constructed from a combination of broadcast parameters and the required content type.

`/topic/<broadcast parameters>/<content type>`

The `content type` segment must be specified, with a value of either `image` or `text`. This allows devices that are capable of receiving only text or only images to differentiate between their desired content type. Devices that are capable of both must subscribe to both topics independently.

The `broadcast parameter` segment is based on the bearer of the audio service being consumed and is specified in the following sections of the present document.

It must be assumed that topics are case sensitive. Topics must be entirely in lowercase.

5.3.1 VHF/FM

The broadcast parameters value for a VHF/FM service topic ID is constructed as follows:

`fm/(<ecc>|<country>)/<pi>/<freq>`

The parameters are populated with the following values:

Parameters	Description	Value	Status
`ecc`	**Extended Country Code (ECC) and country code** Service broadcast RDS [7] ECC concatenated with the first character of the broadcast RDS PI code (country code) must be supplied if available.	*3-char hexadecimal*	*mutually exclusive*
`country`	**ISO 3166 two-letter country code** In the event that a service broadcast ECC is unavailable, an ISO 2-letter country code [8] must be provided.	*2-char string*	
`pi`	**Programme Identification (PI)** Service broadcast RDS PI code.	*4-char hexadecimal*	*mandatory*
`frequency`	**Frequency** Frequency on which the service broadcast is received, formatted to 5 characters in units of 100KHz. Frequencies below 100Mhz must be supplied with a leading zero, for example 95.8 would be represented as 09580, 104.9MHz as 10490.	*5-char string*	*mandatory*

A server must implement topics for both `ecc` and `country` values to handle a situation where the device may not acquire the RDS ECC.

For a detailed explanation on these parameters and their values, please refer to the VHF/FM section of RDNS01 [2].

5.3.2 DAB/DAB+ Digital Radio

The broadcast parameters value for a DAB/DAB+ Digital Radio service [9] topic ID is constructed as follows:

`dab/<ecc>/<eid>/<sid>/<scids>/[(<appty-uatype>|<pa>)]`

The parameters are populated with the following values:

Parameters	Description	Value	Status
`ecc`	**Extended Country Code (ECC)** Service broadcast multiplex ECC code.	*3-char hexadecimal*	*mandatory*
`eid`	**Ensemble Identifier (EId)** Service broadcast multiplex ensemble ID code.	*4-char hexadecimal*	*mandatory*
`sid`	**Service Identifier (SId)** Service broadcast identifier.	*4 or 8-char hexadecimal*	*mandatory*
`scids`	**Service Component Identifier within the Service (SCIdS)** Service broadcast component identifier within the service.	*1 or 3-char hexadecimal*	*mandatory*

If the audio service is delivered as data via X-PAD, the following additional parameter is mandatory:

Parameters	Description	Value	Status
`appty-uatype`	**X-PAD Application Type (AppTy) and User Application type (UAtype)** The X-PAD Application Type number and User Application Type, concatenated with a hyphen (only for applications broadcast in X-PAD). Where Application Types are allocated in pairs, the lower value (indicating the start of the application data group) must be used.	*2-char hexadecimal, hyphen, 3-char hexadecimal*	*mandatory, when referring to an X-PAD component, otherwise omitted*

If the audio service is delivered as data in an independent Service Component, the following additional parameter is mandatory:

Parameters	Description	Value	Status
`pa`	**Packet Address** Packet address of the data service delivering the audio service.	*integer, between 1 and 1023*	*mandatory, when referring to a data service component, otherwise omitted*

For a detailed explanation on these parameters and their values, please refer to the DAB/DAB+ section of RDNS01 [2].

5.3.3 Digital Radio Mondiale (DRM)/AM Signalling System (AMSS)

The broadcast parameters value for a DRM [10] / AMSS [11] service topic ID is constructed as follows:

`(drm|amss)/<sid>`

The parameters are populated with the following values:

Parameters	Description	Value	Status
`sid`	**Service Identifier (SId)** Service broadcast identifier.	*6-char hexadecimal*	mandatory

For a detailed explanation on these parameters and their values, please refer to the DRM/AMSS section of RDNS01 [2].

5.3.4 iBiquity Digital Corporation's HD Radio™ (HD Radio™)

The broadcast parameters value for a HD Radio™ [12] service topic ID is constructed as follows:

`hd/<cc>/<tx>`

The parameters are populated with the following values:

Parameters	Description	Value	Status
`tx`	**Transmitter Identifier** Service broadcast identifier	*5-char hexadecimal*	*mandatory*
`cc`	**Country Code** Service broadcast country code	*3-char hexadecimal*	*mandatory*

For a detailed explanation on these parameters and their values, please refer to the HD Radio™ section of RDNS01 [2].

5.3.5 IP-delivered audio service

At the time of publication RDNS01 [2] does not currently have a defined method of identifying an authoritative FQDN for an IP-delivered audio service. As a result, the present document does not define a method for obtaining visual content for IP-delivered audio services.

Future versions of the RadioDNS specification will define signalling for IP-delivered audio services and this document will be updated accordingly.

5.4 Receiving a message

Once subscribed to a topic, the client will receive messages in the following format:

```
MESSAGE
destination:/topic/foo
message-id: <message-identifier>
content-length: <body byte length>

<body>^@
```

The destination header confirms which topic the message has been received from to help differentiate when subscribed to multiple topics.

The `content-length` header is recommended within the Stomp specification as a byte count of the message body minus the terminator, but within RadioVIS is recommended only as a check. Parsing must end when a null terminator occurs, regardless of the `content-length` header value. If the `content-length` is missing then no such check should be performed.

The body of a message contains messages specific to the RadioVIS protocol. These are defined in section 7 'Message Bodies' of the present document.

6. Comet transport

Comet is a collective term for a number of technologies commonly used together to achieve the impression of pushing data to a client over the pull-orientated HTTP protocol [13].

This is achieved via 'long polling'. In the example of RadioVIS, in place of subscribing to a topic via Stomp, an HTTP request is made which is then held open until a new message frame should be delivered, when a response is sent and connection is closed.

6.1 Request

As opposed to connecting to a Stomp server and subscribing to a topic, a simple HTTP request is made to the server obtained via the initial RadioDNS lookup. Please refer to section 4.1 'Application Discovery' of the present document and RDNS01 [2] for detailed information.

The request URI is constructed as follows:

```
/comet?topic=<topic>[&topic=<topic>[&topic=<topic>...]]&last_id=<last_id>
```

The parameters are populated as follows:

Parameters	Description	Value	Status
`topic`	**Topic** Topic(s) this client wishes to subscribe to. See the Stomp transport implementation (Section 5.2 of the present document) for details on construction details. Each topic should be provided as an individual query parameter.	*string*	*mandatory*
`last_id`	**Last message ID** Provide the `RadioVIS-Message-ID` HTTP header value from the last RadioVIS Comet response. This parameter must be omitted in the first request for the service.	*string*	*mandatory for each request after the initial response, otherwise omitted*

All query parameter values must be URL-encoded [15]. For example:

```
/comet?
      topic=%2Ftopic%2Ffm%2Fce1%2Fc586%2F09580%2Fimage&
      topic=%2Ftopic%2Ffm%2Fce1%2Fc586%2F09580%2Ftext
```

When making the request to the HTTP server, the connection will be held open until such a time that a new message is sent to any of the requested topics. When this occurs, a response will be sent and the connection be closed in the same way a normal HTTP transaction is handled. A timeout of 300 seconds or more should be employed on such a connection as it may be many seconds before the next response is received.

When a response is returned, another request should be made immediately to collect the next message.

The `RadioVIS-Message-ID` HTTP header from the last received message should be included as detailed above. If the server identifies that messages have been sent after the message identified by `last_id` in the request, the server must immediately send the first missed message.

If more than one message has been missed, when the client reconnects again and provides the first missed message ID, the server will immediately return the next and so on until the client has caught up with messages on the server.

6.2 Response

The response will be in the format of a standard HTTP response, with the addition of a RadioVIS-Message-ID header.

```
HTTP/1.1 200 OK
Date: Sun, 13 Sep 2009 15:44:59 GMT
Content-Length: 61
Keep-Alive: timeout=15, max=91
Connection: Keep-Alive
Content-Type: text/plain; charset=UTF-8
RadioVIS-Message-ID: msg-#stompcma-29608762

TEXT Now Playing on Heart: Starship - Nothing's Gonna Stop Us Now
```

The server may use the `message-id` value from the relayed Stomp message for the `RadioVIS-Message-ID` header or any other value that is suitably unique for each response.

The body of the HTTP response is identical to the frame body of a Stomp message and can be processed in the same way.

7. Message bodies

7.1 TEXT message

Provides a text message to be displayed on the device.

```
TEXT <message>
```

This message must be displayed immediately and will replace any existing text message on the client.

A text message must be no longer than 128 characters.

7.2 SHOW message

Provides an HTTP URL to a slide image.

```
SHOW <url>
```

Optionally, a trigger time and/or an associated link URL may be provided via the message frame headers (for Stomp) or via the HTTP response headers (for Comet).

```
trigger-time: <trigger time>
link: <url>
```

The slide associated with the URL provided must be of either JPEG [16], PNG [17] or animated PNG (APNG) [18] format. It is recommended that slides be of dimensions 320 pixels wide by 240 pixels high (see Annex A for detailed information on slide formats).

It is strongly recommended that devices implement a cache to store downloaded slides (see Annex A).

When a SHOW request is received the device must check the cache to see if the slide, uniquely identified by its URL, has already been downloaded. If the asset does not exist in the cache it must request and store the slide, identified by its URL. Each unique slide must be served from a unique URL. For example, a change in an existing slide's content must be served from a new unique URL so devices download the updated slide.

The `trigger-time` value determines when and if the slide is shown. If the `trigger-time` value is specified as 'NOW', then the slide must be shown immediately. If no `trigger-time` value is specified or the time is historical, the slide is loaded to the cache and not displayed. Any other `trigger-time` value should be interpreted as an ISO 8601 combined date and time representation [14] and the slide must be stored in the cache and not shown until the specified date and time has been reached. If there is no cache available on the device, any slide with a `trigger-time` value other than 'NOW' must be ignored.

If a SHOW message is received for a slide already cached, the trigger time for the cached slide must be updated to reflect the new value. If a SHOW message is received for a slide that is currently being displayed, it must not trigger a refresh on the device display but the cached slide trigger time must still be updated.

The `link` value must be a valid URL and can be used by the device to provide associated content when the slide is interacted with. The URL must only be an HTTP-based resource and should be an (X)HTML document that can be rendered in a browser on the device. The maximum length of the URL is 512 characters [15].

Annex A: Acceptable device implementations

RadioVIS has been designed to mirror the specification of DAB Slideshow as closely as possible and allow an implementation of this on other broadcast protocols as well as with IP-delivered audio content. To this end, it is specified that slide contents delivered using RadioVIS must adhere to the DAB SlideShow application specification [1].

Specifically, executions of RadioVIS must adhere to the DAB SlideShow enhanced profile specification, which states:

- "Receivers are strongly recommended to implement a display equal to or larger than 320 × 240 pixels, at a colour depth of at least 15 bits per pixel. Receivers shall not implement SlideShow on displays smaller than 160 × 120 pixels."

- "The SlideShow application display may be rotated to best fit the physical display aspect ratio (portrait or landscape), assuming that the majority of content will be formatted to fit a landscape display. However the orientation of the SlideShow application display shall be consistent across all services, and individual images received by the application shall not be rotated on a case-by-case basis."

- "The original aspect ratio of the image shall always be preserved."

- "Images may be scaled at factors of 150% or greater in order to maximise the available physical display space."

- "It is mandatory to implement a scale factor of 50%, and this is the only downscaling factor permitted."

- "The use of anti-aliasing and similar techniques is strongly recommended to optimise the quality of the scaled images."

- "All receivers shall be able to decode images up to ... 450 kbytes (460 800 bytes)."

- "The Holding Buffer shall be at least 450 kbytes (460 800 bytes) and be able to store between 1 and 64 images. When multiple images are stored, each image may be a different size and/or colour depth."

History

Document history		
First Release	September, 2008	Initial release
V0.6.1	June, 2009	No changes to Visualisation spec, number jump to align all RadioDNS specification documents
V1.0.0	September, 2009	Final release