DAB OVER IP

Alexander Erk & Fabian Sattler

Institut für Rundfunktechnik (IRT), Germany

ABSTRACT

For the transparent consumption of radio services either via Digital Audio Broadcasting (DAB) or Internet Protocol (IP), the same level of metadata and data services for both of the transmissions is necessary. Otherwise the listener would notice the (even temporary) switch over from DAB+ to IP and back. Rebuilding the same level of audio- and data services compared to DAB, could be difficult in IP delivered scenarios.

This paper presents a technology, to use the DAB Encapsulation of DAB Interfaces (EDI) specification for single service streams as a delivery protocol for IP Radio streaming. The same signal the broadcasters use for the DAB transmission will be reused by a server component which splits the whole ensemble of DAB services into single service EDI streams. Service following becomes easy in the same bitstream, data services and announcements support are included.

The paper includes calculations of the overhead the current technical solution has and gives an outlook how to reduce this overhead and an overview about the future work.

INTRODUCTION

Radio is still an extremely popular medium. Under many different usage scenarios, people can follow a radio program, be provided with news, traffic and weather information, listen to music and simply have fun with it. However, the linear medium of radio faces enormous challenges to the acceptance of listeners. For today's listeners, access to audio content is as natural as turning on the tap. The success of smart mobile devices, mobile IP connectivity and the fact that most audio content is "device-neutral" means that people can hear almost anything, anywhere, at any time. At the same time, the new audio offerings as podcast and streaming services are multiplying explosively and there is more choice available to listeners than ever before.

Many tongues claim that these developments mean the end of linear radio and that listeners' needs are met by non-linear, personalized streaming and downloading services. Is that so?

If you put linear live radio and streaming/on-demand audio next to each other and forget the "war of the worlds..." for a moment, a meaningful combination of these two forms holds enormous potential.

For many of its loyal listeners, radio is a medium that is repeatedly switched on because it entertains, informs and frees them from the burden of choosing and becoming active themselves. It offers something new and surprising as well as a kind of an "emotional home" for the listener. OnDemand and streaming, on the other hand, offer control, (almost) limitless choice and direct feedback on the services. The EU-funded project HRADIO wants to show radio stations exactly at the interface of these worlds how such hybrid services look like and how they can be realized on common platforms (Android, iOS and HTML/JS). The following focal points have been identified: Integration of technology, integration of services and integration of the listener.

The development of radio applications is extremely inhomogeneous and diverse. Application developers need a uniform software interface to radio services in order to concentrate on domain-specific tasks and seamlessly integrate radio and on-demand content. At the same time, broadcasters must merge their offerings (broadcast and non-linear) and understand them as a homogeneous service.

To enable developers to concentrate their work fully on the functionality of the applications, the HRADIO project chose the Open Mobile Radio Interface (OMRI) standard as the API to implement access to radio-specific functionalities. OMRI is an API specification currently available as Java API documentation and was standardized by the WorldDAB Technical Committee and ETSI in 2018. As part of HRADIO's work, the OMRI specification for Android devices with DAB USB receivers was implemented and is used in the project as a basis for development of hybrid radio mobile applications.

However, it's clear, that not all of the envisaged end user devices and platforms will be equipped with DAB hardware receivers or are not reasonably usable with a plugged in receiver device via the USB connector. Therefore, HRADIO applications must be able to utilize pure IP delivery of radio services.

A digital radio service is more than just the stereo audio signal, it is difficult to have a full replacement for DAB signals at the pure IP level. First of all, pure IP streaming services do not have a full replacement to the DAB DynamicLabel or DynamicLabel+ data service. Although there are the so-called ICY tags in Shoutcast[1] streams, they cannot reach the functionality of DynamicLabel+ messages and tags. Furthermore, IP Streamed Audio does not offer any alternatives for the slideshow services in DAB, nor does it signal service following and alarm announcements.

These services and signalling surely can be replaced in own application developments by proprietary solutions, but this means a considerable additional development effort as well as additional infrastructure and server capacities. Even if such proprietary solutions exist, they are usually realised over additional IP connections (STOMP, WebSockets, XMLHTTP requests) and do not provide the same level of synchronisation with the audio service as a self-contained system such as DAB.

Therefore, this document proposes the usage of DAB multiplexes in its standardised Encapsulation of DAB Interfaces (EDI) format for the streaming of point to point IP radio services as a replacement for the commonly used Shoutcast streaming.

---

[1] https://en.wikipedia.org/wiki/SHOUTcast

USING EDI FOR IP STREAMING OF SINGLE SERVICE RADIO

EDI has been developed for the transport of Service Transport Interface (STI) or Ensemble Transport Interface (ETI) data streams for the distribution in IP networks. As depicted in Figure 1, EDI follows a tag length value approach, where a single EDI frame represents a single STI-D or ETI 24ms logical frame. Afterwards these TAG items are grouped into single EDI packets and are passed onto the Distribution and Communications Protocol (DCP) for Application Framing (AF).

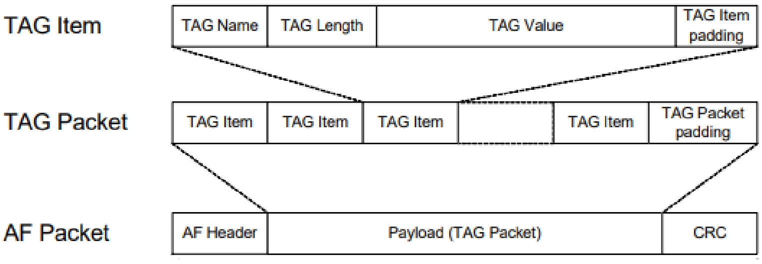

Figure1: AF Packet Hierarchy (EDI)

Usually EDI is used for the contribution of DAB ensemble multiplexes from the multiplexer device to the DAB modulator for transmission. At the multiplexing side, service providers deliver all necessary audio, textual and visual content items as well as the necessary signals such as alarm announcements, emergency warnings or service following to the multiplexer. The multiplexer assembles the ETI stream and performs the packetization into the EDI TAGs and AF frames which are then transmitted over IP networks to the modulation side. The modulator itself can reassemble the ETI stream and finally transmits the DAB digital radio signal modulated as Coded Orthogonal Frequency Division Multiplexing (COFDM) into the air.

Our approach takes the full EDI ensemble multiplex and performs a re-multiplexing of the contained digital radio services into single service EDI streams. These single service EDI streams then are delivered as HTTP or HTTPS chunked transfer as usually used in Audio IP streaming.

EDI Remultiplexer

The EDI remultiplexer solution, written in the Go programming language, uses the full DAB ensemble stream consisting of multiple DAB programme and data services received via UDP unicast or multicast stream.

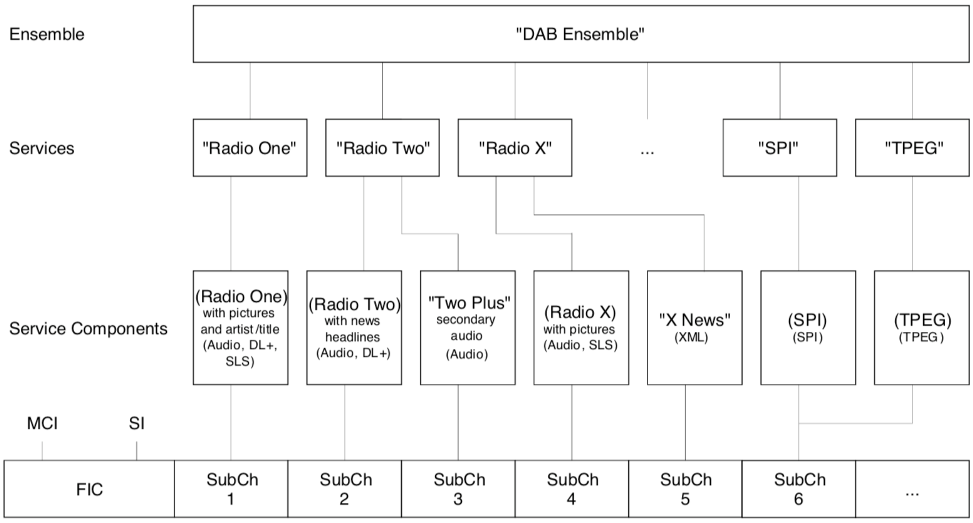

Figure 2: Full DAB ensemble structure (DAB)

The software decodes the minimum necessary Fast Information Groups (FIG) to reconstruct the ensemble. After the ensemble information was decoded each service is extracted and its belonging FIGs are rebuilt in a way to contain only the information needed for this single service.
Table 1. lists the extracted FIGs for a DAB programme which need to be altered by the EDI remultiplexer in order to deliver a standard conform single service EDI stream out of the multi service input stream.

Fast Information Group	Description
FIG 0/1	Basic Subchannel Organization
FIG 0/2	Basic Service Component Definition
FIG 0/8	Service Component Global Definition
FIG 0/13	User application Information
FIG 0/14	FEC Subchannel Organization

Table 1: Extracted Fast Information Groups

All other important FIGs, like Ensemble and Service Labels in FIG 1/0 and FIG 1/1, contained in the Fast Information Channel (FIC) are not altered at all and are copied into the single stream FIC.

The FIGs for traffic announcements will be inserted into the EDI stream on demand. As only few FIGs have to be adjusted for the single EDI stream it is very lightweight and efficient.

After all necessary information is received and the single service EDI streams are reconstructed, the EDI remultiplexer opens a HTTP server with a REST-like interface to receive the EDI stream service in the form like *http://domain:port/services/<subchannelId>*. Thus, this also makes it simple to integrate it into reverse proxy infrastructure for secure HTTPS delivery of the streams.

Designed as a microservice, the EDI remultiplexer will perfectly scale and fit into cloud infrastructures. Figure 3 shows all the involved components. In the left-hand side, the traditional DAB+ is assembled and used for COFDM Modulation in BAND III. This EDI ensemble is forwarded also to the EDI remultiplexer, which generates the single service streams for distribution over IP connections. Hybrid receivers on the right only see DAB+ services with all of its components.

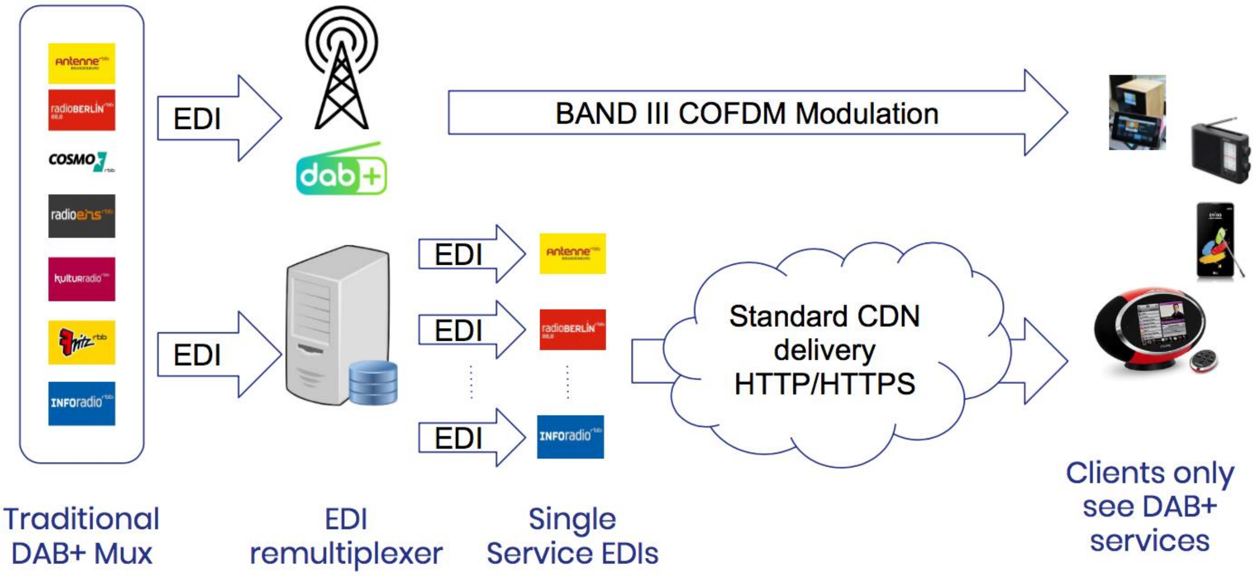

Figure 3: Full broadcast and IP streaming data flow for the DAB over IP solution

Overhead Considerations

This overhead consideration compares the additional overhead from the EDI format to a traditional Shoutcast ADTS AAC and MP3 stream. It compares streams with a bit rate of 128 Kbit/s and 48000 Hz sampling rate each without any additional in-band metadata like Dynamic Label and Extended Programme Associated Data (X-PAD) user applications like Slide Show for DAB EDI streams or ICY tags for Shoutcast streams. As all compared streaming formats use HTTP for transmission it is left out of the comparison.

The values are rounded up to two positions behind the decimal point.

MP3 stream format

The MPEG-1 Audio Layer-3 (MP3) format consists of a 4 byte long header with an optional 2 byte CRC. With 1152 samples per audio frame and a sampling rate of 48 kHz, one audio frame contains 24 ms of audio data.

Without the CRC, the MP3 format has an overhead of 1333,34 bits per second and 2000,00 bits per second with the optional Cyclic Redundancy Check (CRC).

Resulting in an overhead of 1,03% without and 1,54% including the CRC.

AAC ADTS format

The Audio Data Transport Stream (ADTS) format consists of a 7 byte long header with an optional 2 byte CRC. With the usual 1024 samples per audio frame and a sampling rate of 48 kHz, one audio frame contains 21,33 ms of audio data.
Without the CRC the ADTS format has an overhead of 2625,28 bits per second and 3375,36 bits per second with the optional CRC.
Resulting in an overhead of 2,01% without and 2,57% with CRC.

EDI DAB format

Each EDI AF frame transports one logical DAB frame and contains 24 ms of audio data. The following table lists the mandatory EDI TAG items which shall be included in every standard compliant EDI AF frame together with its description and the bytes needed for the TAG item.

TAG Name	Description	Length (bytes)
*ptr	Protocol type and revision	16
deti	DAB ETI Management	14 - 121 (variable)
est<n>	ETI Sub-Channel-Stream	11 + payload

Table 2: Mandatory EDI TAG items

The deti TAG item has a variable length depending on the flags contained within in the deti header. It may extend the minimum length with 3 optional fields:

- Absolute Time Stamp (ATST) field with 8 bytes
- Fast Information Channel (FIC) field with 96 bytes
- RFU Data field with 3 bytes

The *est*<*n*> TAG item consists of 11 bytes header information following the DAB Main Service Channel (MSC) subchannel payload. The overhead of the payload itself depends on the type of the DAB programme service. For the MPEG-1 Audio Layer-2 format used in DAB, 4 bytes of MPEG header, 2 bytes CRC and 2 bytes of the fixed PAD field are added. DAB+ uses AAC audio encoding and so-called audio super framing to provide a synchronization mechanism. Each DAB+ Audio super frame consists of 5 consecutive DAB frames and therefore contains 120ms of audio data.

The audio super frame is synchronized with the so-called header fire code of 2 bytes indicating a correctly built audio super frame. Each audio super frame has a header of 1 byte. Depending on the AAC SBR encoding option and the used sampling rate an audio super frame contains 2, 3, 4 or 6 AAC audio frames each with an additional 2 byte CRC protection.

These mandatory TAG items are packed into a surrounding Application Frame (AF) adding 10 bytes of header information and a 2 byte CRC at the end of the frame.

For a DAB EDI stream with MPEG-1 Audio Layer-2 the complete overhead from the EDI format and the audio payload itself sums up to 52333,28 bits per second. Resulting in an overhead of 29,02%.

For a DAB+ EDI stream with AAC audio the complete overhead sums up to:

- 51000,00 bits per second for 6 audio frames per audio super frame
    - Resulting in an overhead of 28,49%
- 50533,28 bits per second for 4 audio frames per audio super frame
    - Resulting in an overhead of 28,31%
- 50266,64 bits per second for 3 audio frames per audio super frame
    - Resulting in an overhead of 28,20%
- 49933,28 bits per second for 2 audio frames per audio super frame
    - Resulting in an overhead of 28,06%

Format	Overhead (%)
MP3 w/o CRC, with CRC	1,03%, 1,54%
AAC ADTS w/o CRC, with CRC	2,01%, 2,57%
EDI DAB MPEG-1 Audio Layer-2	29,02%
EDI DAB+ AAC 6, 4, 3, 2 AUs	28,49%, 28,31%, 28,20%, 28,06%

Table 3: Format overhead comparison

FUTURE WORK

As we have seen in the previous section, the EDI format adds a significant amount of overhead when used for single service point to point IP distribution. However, first calculations show a great potential to reduce overhead. In the connection-oriented TCP distribution of EDI services it is sufficient to transfer the FIC only in the first packets until all, most static, data is delivered. After that, only when changes to the service state occur, like traffic announcements, the FIC contained in the *deti* item would be transferred again. Also, the **ptr* item, which indicates the protocol type and its revision, could be transferred only in the first EDI-AF-frame. By transferring the necessary tags only in the first few frames the overhead could be reduced to only 6,57% overhead.

The tag, length, value approach used by EDI also opens up for proprietary extensions without losing compatibility to other standard receivers. As per definition, all unknown tags shall be ignored by the receiver. Thus, enabling richer and personalized user experiences within the same, standard compliant stream. To make these changes happen, work has to be done in the according working groups with WorldDAB.

The low delay AAC encoding used in DAB+ with 960 samples per audio frame is currently problematic for Web-audio players. These players expect as default an AAC stream with a 1024 MDCT window, resulting in a down-pitched audio playback. Also, the ADTS streaming format, used by most Web-players, doesn´t allow to signal the frame length. We are working on this to find solutions together with partners from W3C.

For a real scalable IP distribution of EDI further work must be done to integrate this format in existing CDN infrastructures, including monitoring and redundant fail-over solutions.

CONCLUSION

The EDI format in its current definition as a broadcast format adds a large amount of overhead compared to Shoutcast MP3 and AAC-ADTS streaming. However, the EDI format backed by an ETSI standard offers a real self-contained radio service with all the features the industry proven DAB broadcast has to offer. Especially the emergency warning and traffic announcement features defined in the DAB standard add real value. Radio service providers benefit from the single playout toolchain for broadcast and IP delivery. Making it obsolete to integrate and service different toolchains and workflows. They can concentrate to create one format for all distribution channels. Leading to a better, feature rich service and listener experience, with the same service quality on all distribution ways. Using the identical data for broadcast and IP distribution also reduces the delay of IP delivered radio services, introduced by different encoding toolchains, to a minimum. In the current lab tests the single service EDI stream over IP, generated by a DAB multiplexer for broadcast distribution, is ahead of the broadcast distribution. By having the same data distributed by broadcast and IP it´s not just a seamless but a bit-perfect service-following experience.

REFERENCES

1. OMRI - ETSI TS 103 632: “Digital Audio Broadcasting (DAB); Open Mobile Radio Interface (OMRI); Application Programming Interface (API)”
2. DAB – ETSI EN 300401: “Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers”
3. DCP - ETSI TS 102 821: "Digital Radio Mondiale (DRM); Distribution and Communications Protocol (DCP)"
4. STI - ETSI EN 300 797: "Digital Audio Broadcasting (DAB); Distribution interfaces; Service Transport Interface (STI)"
5. ETI - ETSI ETS 300 799: "Digital Audio Broadcasting (DAB); Distribution interfaces; Ensemble Transport Interface (ETI)"
6. EDI - ETSI TS 102693: “Digital Audio Broadcasting (DAB); Encapsulation of DAB Interfaces (EDI)”