Integration of a fully hybrid and multiplatform Digital Radio chain

Mathias Coinchon
EBU TECHNICAL

Stanislas Roehrich
EBU TECHNICAL

Michael Barroco
Swiss Institute of Technology, Lausanne (EPFL)

The EBU, in partnership with CRC from Canada and Global Labs from the UK, has integrated a full digital radio chain that is hybrid and multiplatform. The objective is to demonstrate a functional radio chain from production through to the listener, using a hybrid broadcast / broadband delivery platform for radio in Europe.

This article gives a technical explanation of how the different elements have been integrated. Most of these elements are available in free open source for further development, or for use by broadcasters or the industry.

In early March 2011, the platform was demonstrated at the Geneva Motor Show.

Leading on from a demo at IBC 2010, the EBU has now integrated a full digital radio chain — hybrid and multiplatform — in partnership with the Communications Research Centre (CRC) from Canada and Global Labs from the UK. The objective is to demonstrate a functional radio chain from production through to the listener, using a hybrid broadcast and broadband delivery platform for radio in Europe. It produces the following radio signals:

- ***DAB/DAB+ visual radio*** using MOT Slideshow;
- ***RadioDNS*** with FM and ***DAB/DAB+*** using the (RadioDNS) RadioVIS specification;
- ***DRM*** using Slideshow;
- ***Internet streaming*** according to the IMDA specification.

Different software radio solutions have been used to produce these signals and could all be integrated on a single standard PC.

The result has been demonstrated simultaneously on *standard profile* receivers (simple radios, without a colour screen) and *advanced multimedia* receivers (with a colour screen, interactive control) such as smartphones.

In order to produce the content, a simple visual radio production platform has been designed, integrating visual elements directly in synchronization with the audio. The production platform dis-

patches the content to the different delivery platforms, demonstrating the concept of “produce once, distribute to many”.

This article give a technical explanation of how the different elements have been integrated. Most of the elements are available in free open source, for further development, or for use by broadcasters or the industry. An appendix is included, reporting on the demo given at the Geneva International Motor Show in March 2011.

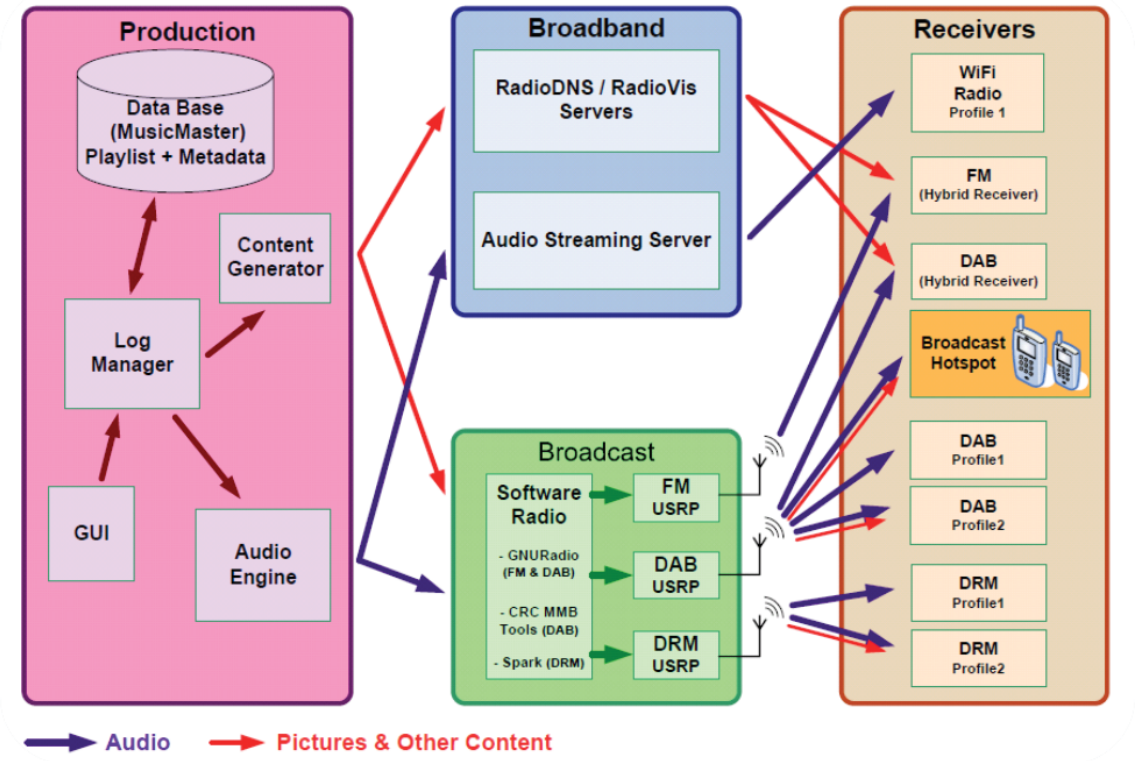

Figure 1
The synoptic and workflow of the demonstration

Production

The objective was to have a flexible production platform in order to integrate easily the visual and metadata elements to the audio, and also in order to adapt the content produced to the different delivery platforms.

It had been decided to create such a platform using a modular approach, enabling further development and a distributed approach for production. Development has been made using C# on a Windows 7 platform – with the benefit of re-using the BASS audio library and the WPF visualization library.

A functional description of the different modules is as follows *(also see Fig. 2)*:

- ***Database for audio/metadata and playlist manager:*** For the IBC demonstration, a commercial database and playlist management system from a commercial company called MusicMaster has been used. However it is adaptable to other databases or systems. A free open MySQL version is currently available as an alternative and a simple front end has been developed to populate the database.
- ***Playout:*** The playout module is the central module managing the live playlist and audio/metadata triggering. The Graphical User Interface (GUI) is integrated with the playout. It is the main screen for the user to load and control the playlist, and to control the visual elements, for example to trigger the insertion of custom visuals coming from webcam snapshots.
- ***Audio engine:*** The audio engine is remotely controlled by the playout module and is signalled from the audio file to start playing the audio. The output consists of either audio to the sound-

card or, directly, an mp3 stream source for an Icecast / SHOUTcast server (in development). It works independently and if other blocks crash, it continues to play the current audio file and can reconnect later to the playout module.

- ***Content manager:*** This module is in charge of aggregating the visual and metadata content coming from the different sources. It renders visual slides on the fly and dispatches them on to the different delivery platforms (mainly via FTP and send signalling via STOMP protocols).

Communication between the blocks is made using WCF (on TCP). This approach will make it possible to distribute elements to different locations and ease any further developments or adaptations.

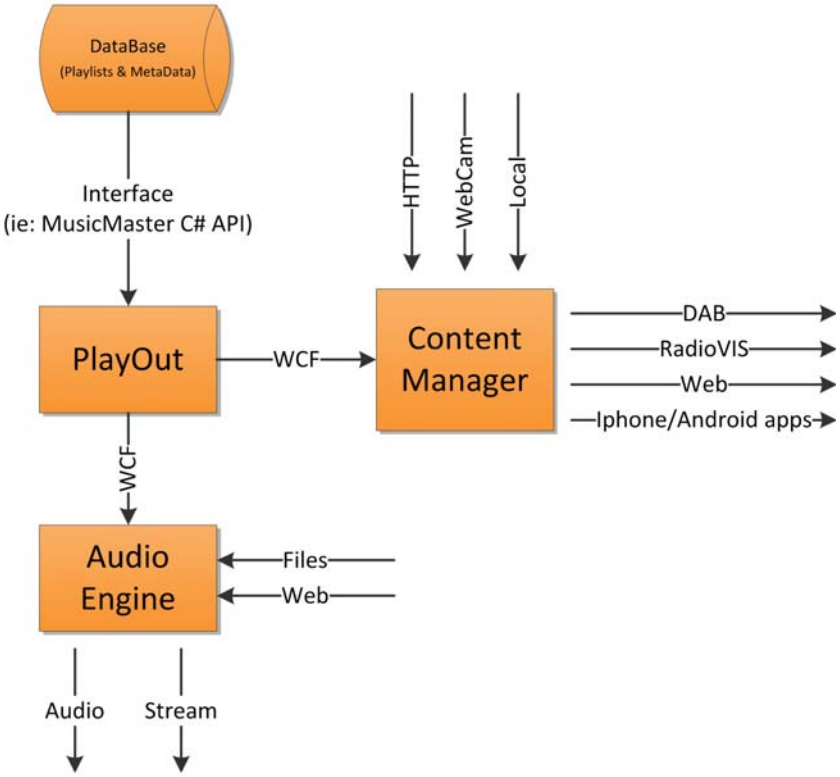

Figure 2
Synoptic of the functional blocks of the production platform

The source code of the platform is available at: **http://code.google.com/p/ebu-radio-production/**

An LPGL licence is used, allowing integration with non-free-project products. The code is available for developers who would like to reuse blocks and integrate them with other production platforms. For example, it could easily be integrated with the WinRadio production platform to generate visual radio at a special event. However, it is important to mention that the code is provided without documentation and support. This is not a turnkey solution for radio production.

Broadcast delivery

The equipment usually required to perform live digital radio transmission is heavy and expensive. In this demo we have used a software-defined radio (SDR) approach – using free or open source developments where available. In this approach, instead of using dedicated equipment for each task (encoding, multiplexing, modulation...), everything is done in software running on standard PC processors. With the power of current PCs, it is possible to perform the following things on a single machine running Linux:

- DAB/DAB+ encoding, multiplexing and modulation using CRC mmbTools, a free open source set of tools developed by Communication Research Center, Canada.
- DRM encoding and modulation using Spark, a software-defined DRM/DRM+ transmitter that has a free version and a professional paid-for version.
- FM Stereo RDS transmission, built from the GNU Radio open SDR framework.
- MP3 streaming using Icecast free open-source HTTP streaming server and SHOUTcast (free but not open source) server.

The output of a software-defined radio system is generally a complex baseband signal (I/Q samples). This I/Q signal is then transmitted into a physical signal by a generic software radio peripheral

that upconverts it and transmits it at the wanted frequency. For this, we used the Universal Software-defined Radio Platform (USRP) from Ettus which is an open-source design (schematics, FPGA code publicly available) selling for $700 (+ $450 for the RF front end).

The output of this box produces a small power signal that can be received locally. However we wanted to demonstrate that this solution can be used for higher-power transmission, to achieve coverage of the whole IBC hall we were in (and its surroundings).

Local DAB transmission

As explained in the previous section, we have integrated CRC mmtools to perform a licensed local DAB transmission.

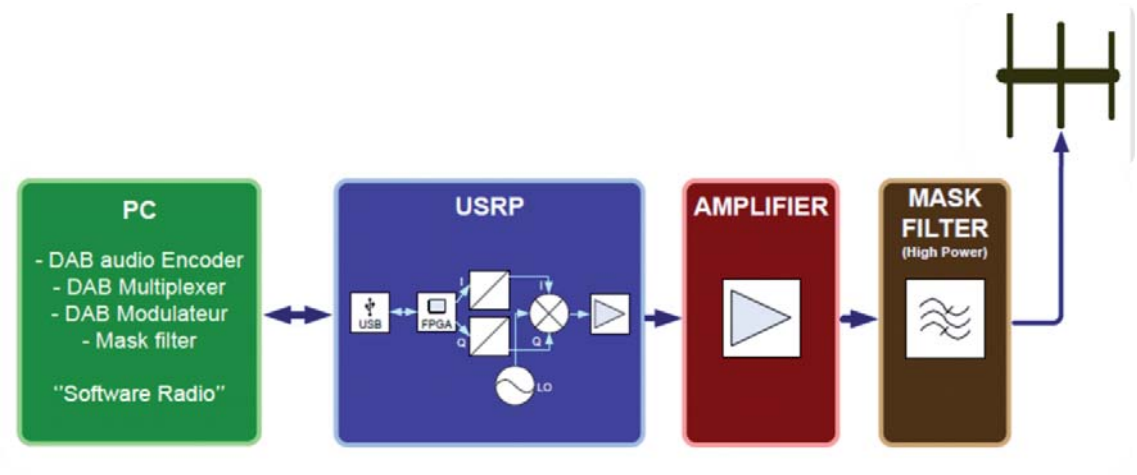

Figure 3
"DAB in a box" hardware configuration to achieve local DAB/DAB+ coverage

Using this generic hardware made it possible to achieve the transmission at extremely low costs:

PC	~ 800 €
Linux, gnuradio, CRC-mmbTools	0 €
USRP + WBX RF frontend	1150 $ (~ 820 €)
Amplifier based on Mitsubishi Power module, 35 Watts CW, 6W OFDM	~ 150 €
VHF Mask Filter from Delta Meccanica, 6 cavities	1300 €
VHF 5dB 3 elements Antenna	300 €
Small equipment	100 €
TOTAL	**~ 3500 €**

On the software side, the following elements have been used *(see Fig. 5)*:

- ***Jack*** – a virtual sound server that permits routing of signals from the soundcard but also generates virtual audio sources (such as streams) and enables the insertion of audio effects (dynamic processing).
- ***Toolame*** – a free open source MPEG Layer II encoder.
- ***CRC-DABPLUS*** – an HE-AAC encoder for DAB+, developed by CRC. We also used the free CRC Slideshow tool to produce the MOT slideshow packet-mode channel.
- ***CRC-DABMUX*** – the free open source DAB, DAB+ and DMB multiplexer. It produces a compliant ETI stream that can be streamed, recorded or sent to an E1 card to feed a DAB COFDM modulator.

- ***CRC-DABMOD*** – a free open source DAB modulator that produces an 16bits I/Q sample stream output.
- The ***baseband player*** has been integrated using GNU Radio to control USRP for the frequency, output power, etc.

Figure 4
DAB in a box rack

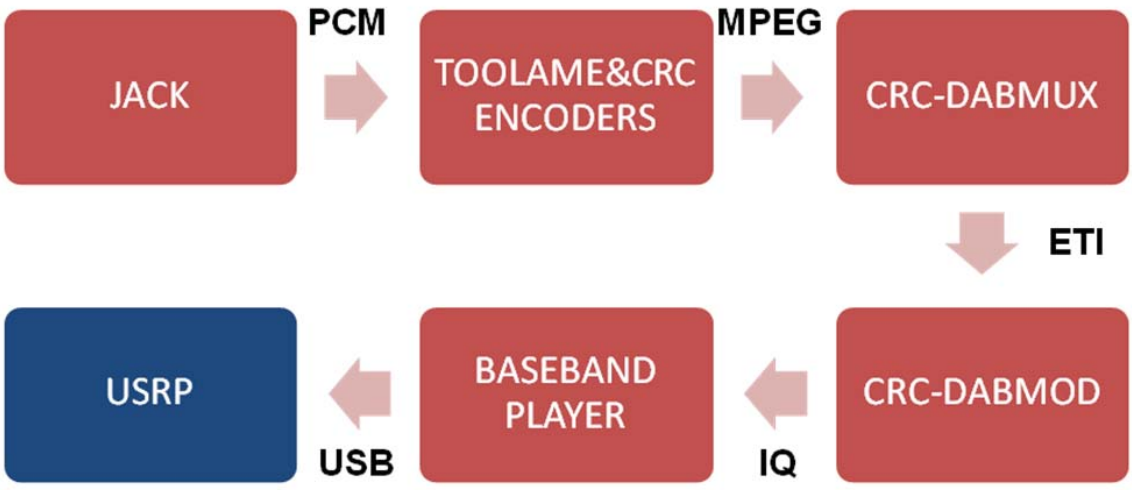

Figure 5
Workflow for the live local DAB transmission

Communication between elements is performed using UNIX pipes.

We have demonstrated here that, thanks to the software-defined radio approach and the free open source tools made by CRC, it is now possible to perform a digital transmission at very low costs. This opens a new field of possibilities for experimentation, the demonstrators and very local transmission. It is important to mention that is it not a commercial solution. It is currently experimental, it requires expert knowledge to run it and no support is provided. However, such an approach using generic hardware and software-defined modulation on PC platforms may become available as commercial products in the future, lowering the infrastructure costs.

We have also demonstrated the benefits of open source solutions for easily integrating the blocks and developments from different parties.

Broadband delivery

Streaming

Broadband delivery has been performed using SHOUTcast (free) and Icecast (free open source) servers running on the same machine, performing software-defined radio.

RadioDNS/RadioVIS

RadioDNS / RadioVIS delivery consists of two parts:

- Content delivery using standard HTTP (production platform).
- Signalling of content using the STOMP protocol that notifies receivers of new slides, text updates and the URLs associated with the slides. RadioDNS allows an alternative protocol, called COMET, to perform the same tasks.

FTP was used to post slides on the web server. The web server used for the demo was an Apache2 running on a server at the EBU.

RadioVIS application server has been developed in Java, internally at the EBU. This server supports the STOMP and COMET protocols. It is useful for tests, demos and experimentation but would require further enhancements for regular use.

More recently, we have developed a RadioVIS server, hosted at the EBU, that can be used by EBU Members to start with a simple visualization service for their broadcast stations *(Fig. 6)*. This server is controlled via a web interface where Members can define and control the visualization service for their stations and can define a fixed image (RadioVIS fallback) to be displayed at any time when no other content-specific visualization is available.

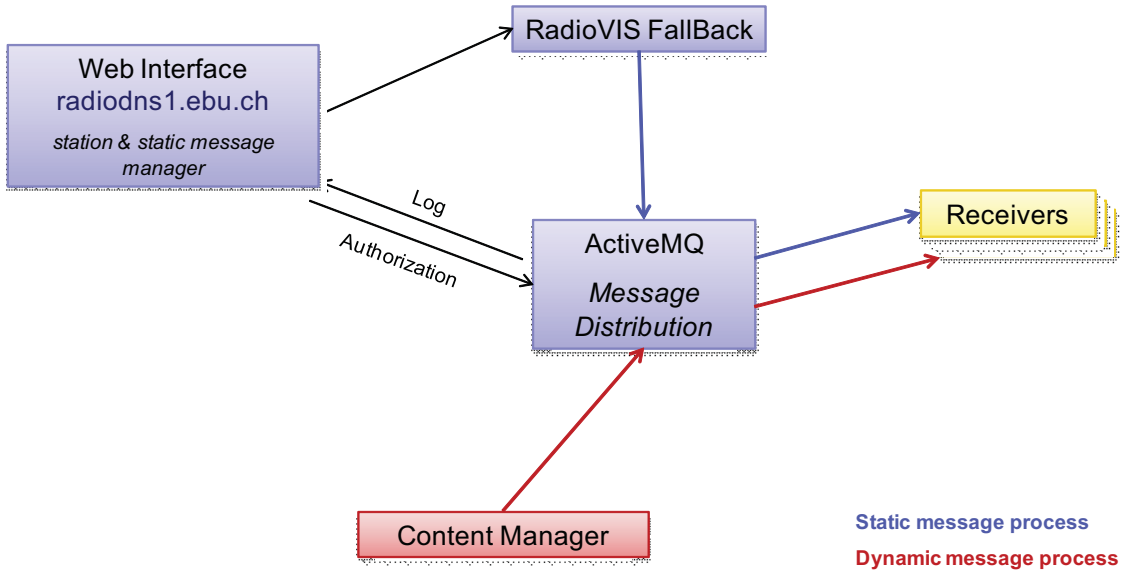

Figure 6
Caption .....

This server also offers broadcasters the possibility of sending RadioVIS message themselves via the server using STOMP protocol. In order to achieve this, they have the possibility to re-use the content manager module from the production platform developed internally *(see previous sections)*.

If you are interested in starting a visual radio service based on RadioVIS for your FM, DAB/DAB+ or DRM stations, then please contact EBU TECHNICAL at:

tech@ebu.ch.

Reception

This multiplatform combination of signals was then displayed on various receivers available from the market *(Fig. 7)*. The objective was to demonstrate a similar radio experience across the different platforms.

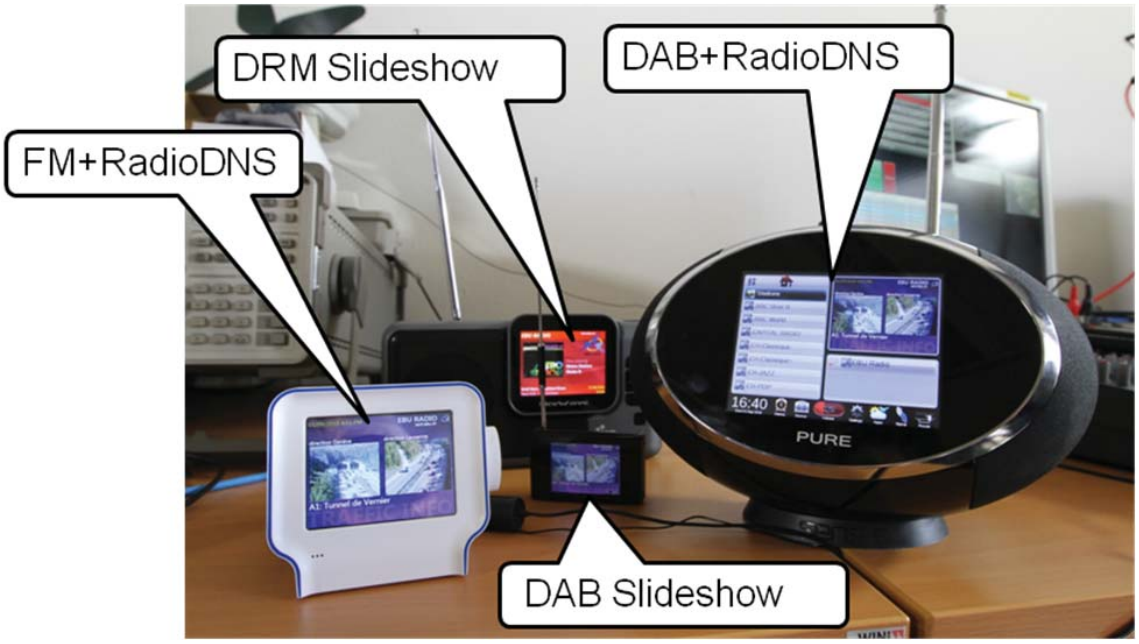

Figure 7
Commercially-available receivers used during the tests

Abbreviations

AAC	Advanced Audio Coding
BASS	(audio library) http://www.un4seen.com/
COFDM	Coded Orthogonal Frequency Division Multiplex
CRC	(Canadian) Communications Research Centre http://www.crc.gc.ca/
CW	Carrier Wave
DAB	Digital Audio Broadcasting (Eureka-147) http://www.worlddab.org/
DAB+	DAB using the AAC codec
DMB	Digital Multimedia Broadcasting http://www.t-dmb.org/ http://www.worlddab.org/
DNS	Domain Name System
DRM	Digital Radio Mondiale http://www.drm.org/
ECC	Extended Country Code
ETI	(DAB) Ensemble Transport Interface
FM	Frequency Modulation
FPGA	Field-Programmable Gate Array
FTP	File Transfer Protocol
GNU	(operating system) http://www.gnu.org/
GUI	Graphical User Interface
HE-AAC	High Efficiency AAC
HTTP	HyperText Transfer Protocol
I/Q	In-phase/Quadrature
IBC	International Broadcasting Convention http://www.ibc.org/
IMDA	Internet Media Device Alliance http://www.imdalliance.org/
LGPL	(GNU) Lesser General Public Licence http://www.gnu.org/licenses/lgpl.html
MOT	Multimedia Object Transfer
OFDM	Orthogonal Frequency Division Multiplex
PI	Programme Identifier
RadioDNS	http://radiodns.org/
RDS	Radio Data System http://www.rds.org.uk/
RF	Radio-Frequency
SDR	Software-Defined Radio
USRP	Universal Software Radio Peripheral http://www.ettus.com/
VHF	Very High Frequency
WPF	Windows Presentation Foundation http://msdn.microsoft.com/en-us/library/ms754130.aspx

In recent times, smartphones and home media platforms have enabled user applications to be developed. This opportunity was used to demonstrate RadioDNS / RadioVIS on platforms with an integrated FM tuner.

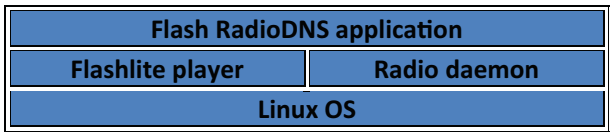

Figure 8
Layers of software in the Chumby One device

Global Labs has demonstrated implementation of RadioDNS with FM on a Nokia N900 phone (Linux-based) and a Sony Xperia phone (Android-based). More recently, CRC in Canada has developed a free library to access the FM receiver on the Samsung Galaxy Android phone. This library will make any RadioDNS application possible on this phone and will become available on the Android Market. Similar developments are also underway for smartphones with FM running the Microsoft mobile operating system. Nokia Symbian phones with a DAB headset could also be used.

Figure 9
Chumby One displaying a RadioVIS service associated with FM

At the EBU, we have created an implementation of RadioVIS / RadioDNS on a Chumby One device. This is a 120$ home multimedia gadget with a colour touchscreen, loudspeaker, Wi-Fi and an FM tuner. The development was possible thanks to the fact that Chumby is fully open source (software *and* hardware). It has been relatively easy to create a RadioVIS application on this device.

The underlying system is Linux and the user applications are running on Flash Lite.

In order to control the FM receiver and get RDS data, a daemon process is running that gets and returns the parameters via HTTP queries and responses. The EBU application consists of:

- A modification of the chumbradiod daemon (written in C) to get the RDS PI and ECC parameters necessary for the RadioDNS query.
- A Flash Lite application that controls the FM receiver and displays RadioVIS slides. However, Flash Lite does not have the necessary calls to perform the special DNS queries required by Radio-DNS and so an HTTP proxy was used to make the DNS query. It is planned to integrate the DNS query in the chumbradiod, to avoid the need for an external proxy.

Conclusions

In this demo a full visual radio chain has been integrated using modern techniques such as software-defined radio, a modular distributed platform such as in Service Oriented Architectures and scalable consumer platforms such as smartphones and multimedia home devices.

EBU TECHNICAL is not intending to sell any products or services but this integration is a demonstrator that can be useful for demos, experimentation and development of new radio services. Everything that has been developed is available to the public for re-use or further development. Since IBC-2010, this demo has been shown at various other shows and events, inluding the recent Geneva Motor Show *(see the Appendix)*, triggering a lot of interest and, we believe, ideas for new radio services. Radio has often suffered from a chicken and egg situation, with the broadcasters and manufacturers waiting for each other to provide new digital or radio services / receivers. Our approach of creating demonstrators, tools and modules, may help to make it happen.

We are also looking to build a community of developers amongst broadcasters and the industry to work on open source developments that are useful for the radio media.

Please get in contact with us if you have radio developers in your organization and don't hesitate to forward this article to them.

More information including source code

The developments created at the EBU have been put in open source. Broadcasters or the industry are free to reuse or enhance these developments:

- EBU Visual Radio production platform: **http://code.google.com/p/ebu-radio-production/**
- EBU RadioDNS / RadioVIS server: **http://code.google.com/p/ebu-radiovis-server/**
- RadioDNS /RadioVIS on Chumby application: **http://code.google.com/p/ebu-radiovis-chumby/**

Mathias Coinchon graduated in 2000 in Communication Systems Engineering from the Swiss Institute of Technology in Lausanne (EPFL), Switzerland, and the Eurecom Institute in Sophia-Antipolis (France). He developed his diploma thesis on Digital Radio Mondiale at BBC R&D, south of London, and then worked as technical project manager at Wavecall, a start-up company that is active in mobile communication network planning.

Before joining the EBU in 2006, Mr Coinchon worked for Swiss Public Radio (RSR) on networks and was then part of the team at SRG-SSR, defining the re-launch of Digital Radio in Switzerland. In his spare time, he is involved in helping a community radio station and runs a website on open source techniques for Digital Radio.

Mathias Coinchon is vice-chairman of WorldDMB TC and actively follows the activities of the Digital Radio Mondiale consortium, Internet Media Devices Alliance (IMDA) and RadioDNS. He is currently working on the harmonization of delivery technologies for Radio in Europe and the hybrid broadcast / broadband approach. In February 2010, he organised a successful Radio Week and Summit at the EBU, gathering these four organisations together to discuss the harmonization of radio delivery. From this event, a proposal emerged to initiate a European Digital Radio Forum to continue collaboration and promote a consistent and unified approach for future radio delivery.

Stan Roehrich received a master's degree in Electronics and Telecom Networks in Switzerland in 1995. He worked for seven years at Actua Films, a broadcast service provider company, where he was responsible for the telecom infrastructure and systems integration, specifically audio, video and RF design development.

In the EBU, Mr Roehrich initially covered majors news and sports events as a broadcast transmission engineer. He then joined Eurovision Business Technologies Unit (BTU) where he handled renewal of the Eurovision control centre infrastructure, network architecture (fibre and satellite) and software definition for network management. Much of his recent work has been on IP-over-satellite systems.

Radio has always held a special place in Stan Roehrich's life and it occupies much of his free time.

Michael Barroco is studying Computer Science at the Swiss Institute of Technology in Lausanne (EPFL), Switzerland. Last summer, he did an internship at the EBU where he developed the production and reception parts of the Hybrid Radio demonstration which was presented at IBC 2010.

Before joining the EBU, Mr Barroco was involved with the student radio station, Fréquence Banane, and worked for a commercial broadcaster as a DJ.

Please note that these developments are not turnkey solutions but tools for further development or experimentation.

Other tools:

- DAB MMBTools from CRC for DAB/DAB+: **http://mmbtools.crc.ca**
- Spark DRM/DRM+ transmitter: **http://www.drm-sender.de**
- Community for experiments, developments on Software Defined Radio: **http://www.opendigitalradio.org**

Credits

This demonstration has only been possible with the work carried out by François Lefebvre and Pascal Charest from CRC in Canada who developed the mmbTools, and Global Labs in the UK for the RadioDNS smartphones.

Appendix
Hybrid Visual Radio at the Geneva Motor Show

The project

At this year's Geneva International Motor Show, held in early March, EBU TECHNICAL demonstrated a hybrid digital radio service in conjunction with Rouge FM radio, a commercial radio group in Switzerland and Swisscom Broadcast, the main operator for radio / TV broadcasting in Switzerland.

Figure 10
Caption ...

Rouge FM had put together a specific digital radio station for the motor show, in a temporary DAB+ multiplex using Swisscom Broadcast infrastructure and VDL equipment. EBU TECHNICAL provided the technical expertise and adapted its open source visual radio production platform to provide a visualization service on DAB+ and RadioDNS.

Swisscom Broadcast has been working with Nokia to adapt their DAB+ headset for Symbian mobile phones in order to display slides associated with radio programmes, making them clickable to follow web links associated with the slides. In addition to this specific programme, other radio stations were present in the motor show multiplex. In particular, car manufacturers who had expressed an interest had their own dedicated channels. These stations could be received in cars fitted with a digital radio, on mobile phones with DAB+ adapters, other portable DAB+ devices and the public could also see the results on domestic radio receivers displayed on a stand.

This trial was an opportunity to show advanced digital radio services at the largest independent motor show in the world. Vehicles are the major listening environment for radio outside of the home, and the availability of digital radio as standard in vehicles is a key enabler for successful digital radio services. Importantly, these digital radio services showcased a commercial broadcast group extending their brand with thematic and event stations as well as providing an enhanced experience through new visualization services.

A short video of the event can be viewed at **http://tech.ebu.ch/news/hybrid-cars-meet-hybrid-radio-15mar11**

Technical infrastructure

For the motor show project, we re-used elements from the IBC demo with the exception of:

- ***The production platform***. Here, the WinMedia platform from Rouge FM was used instead and we adapted the content manager module from our custom production platform to provide visualization. This work could be performed with limited effort, due to the modular approach of the production platform made for the demo and the flexibility of WinMedia.
- ***The DAB+ headend***. A commercial professional infrastructure for the DAB+ headend was used with D-Vaudax multiplexer and D-audience encoders from VDL. Here also it was very easy to dispatch the content from our content manager.

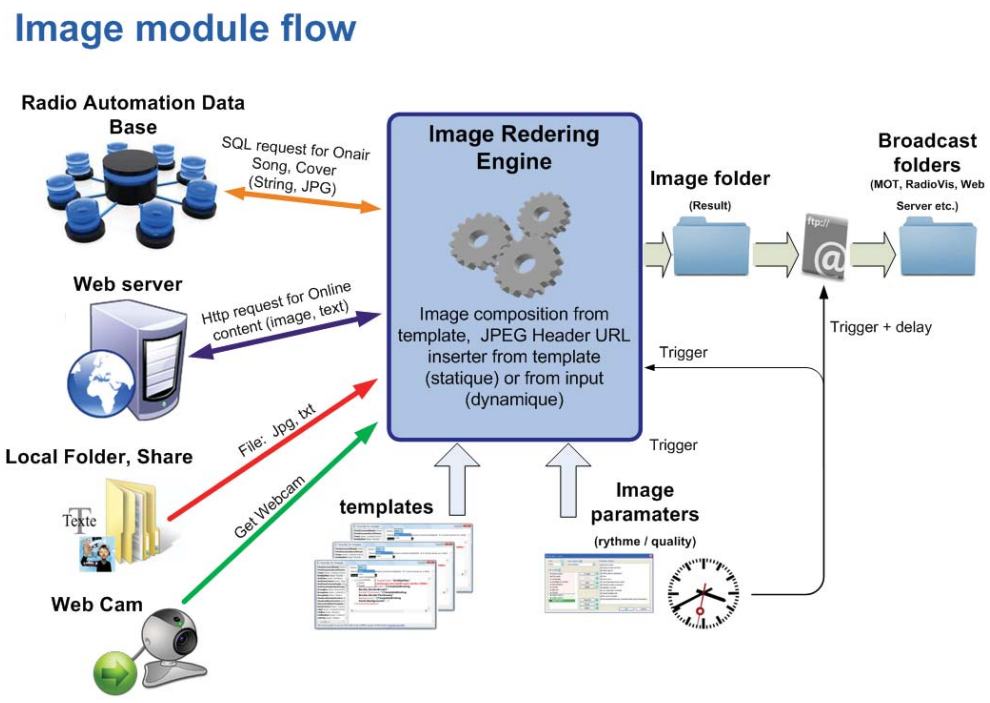

Figure 11
Image rendering and dispatch workflow

The hybrid experience was also possible on Nokia C-7 and N-8 phones equipped with the DAB headset. Nokia had just released the product and couldn't implement RadioDNS in time for the motor show. However as the phone was already decoding XPAD DAB slideshow, a custom solution has been found to associate specific URLs with the slides and so make them clickable.

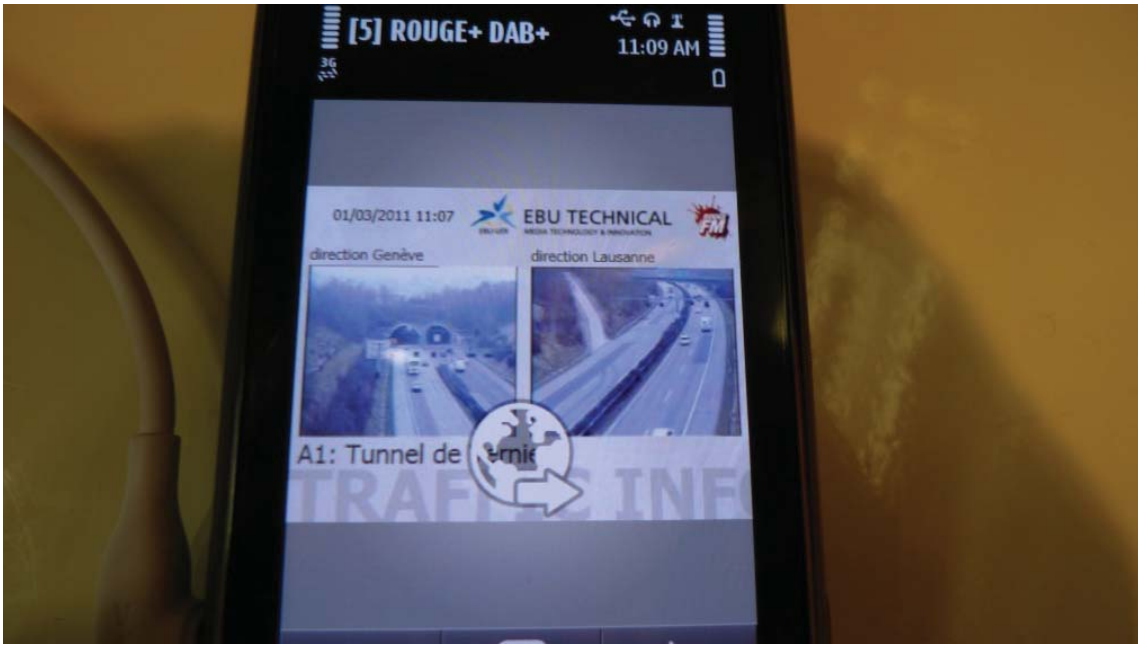

Figure 12
DAB Slideshow display on a Nokia N8 with DAB headset: the Earth in the middle is the clickable link associated with the slide

The text of the URL has simply been added in the "Comment" field of the JPEG image header. This way of doing it proved to be very simple to implement, both on the production and receiver sides (for Nokia). Moreover as the DAB headend transports the images in a transparent way, no modification has been necessary on the encoders or multiplexer. A similar mechanism may be standardized in the future by WorldDMB.

This version: 28 March 2011

Published by the European Broadcasting Union, Geneva, Switzerland

ISSN: 1609-1469

Editeur Responsable: Lieven Vermaele

Editor: Mike Meyer

E-mail: tech@ebu.ch

The responsibility for views expressed in this article rests solely with the authors