IDAG / OMRI

Jorn Jensen, NRK

Dr. Les Sabel, Technical Committee, WorldDAB

Kuala Lumpur, 6 March 2017
31 members from 17 countries

EBU

KONSOLE LABS

IRF

RTE

Radio Wroclaw

BBC

ARS Traffic & Transport Technology

norkring

World DAB

Commercial Radio Australia

Sia Media

Radio.no

RTBF

Fraunhofer IIS

DAB DAB ITALIA sc.p.a.

Digital Radio

digital one

NMTV

MOBILE TV NL

Deutschlandradio

PANEDA

DMB Mongolia
RNI Radio, Latvia
DigiBNetworks, Malta
Mobile TV PTY, South Africa
LG Stylus 2

The world’s first DAB+ enabled smartphone went on sale in over 20 countries in Q2.
When *the USP* is DAB+, that must be clearly communicated.
LG Stylus DAB+

DAB+ ideally needs to be in the name
WorldDAB, the EBU and IDAG have formally **teamed up** to better be able to incentivize more DAB+ enabled smartphones.
Open Mobile Radio Interface
OMRI is
the open and universal
smartphone “bridge”
between
the DAB+ chipset and the apps.

Announced during IBC, 2016.
github.com/ebu/OpenMobileRadioInterface
The trilogy is complete. Part 3 available now.
Now Playing... Dirty Talk, Wynter Gordon

Now playing on 1041 2DayFM: The Bad Touch by Bloodhound Gang
OMRI Technical Approach

Technical approach

- Open API – to be published on Android Developers website (TBC)
  - Currently available as Open Source on EBU gitlab: [https://github.com/ebu/OpenMobileRadioInterface](https://github.com/ebu/OpenMobileRadioInterface)
- Provide Open Source example code
  - To help smartphone manufacturers
  - To help App developers

• Current status
  - Focus has been on LG Stylus 2 DAB implementation
  - Currently using the IRT shim layer between LG API and OMRI API
  - Updates and additions in progress
  - Minimum requirements drafted

New WorldDAB Technical Committee Task Force established – Task Force OMRI
# OMRI – Minimum Requirements

<table>
<thead>
<tr>
<th>Profile</th>
<th>General DAB requirements</th>
<th>Metadata/Data services (User applications)</th>
<th>API Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Core Profile</strong></td>
<td><strong>(mandatory minimum requirement)</strong>&lt;br&gt;Band 3 reception (174 to 240 MHz); Mode 1 operation&lt;br&gt;API permits 'band scan' and 'tune to specific frequency' returning available ensemble(s), services, service components including basic parameters (audio (DAB/DAB+), data (UATy), etc).&lt;br&gt;DAB audio&lt;br&gt;• MPEG layer 2&lt;br&gt;• MPEG-4 HEAACv2&lt;br&gt;One sub-channel with minimum 144 Capacity Units (e.g. 192 kbps@EEP-3A/UEP-3)&lt;br&gt;All FEC code rates (UEP and EEP)&lt;br&gt;Additional sub-channel,&lt;br&gt;• Minimum additional 24 Capacity Units (e.g. 32kbps@EEP-3A)</td>
<td>Text:&lt;br&gt;• Character set decoding&lt;br&gt;  o Complete EBU Latin based repertoire&lt;br&gt;  o UTF-8&lt;br&gt;• Service label and service component label&lt;br&gt;User Applications:&lt;br&gt;• Dynamic Label&lt;br&gt;• Slideshow&lt;br&gt;• Categorised SlideShow&lt;br&gt;• ClickThroughURL&lt;br&gt;• Dynamic Label+&lt;br&gt;Packet Mode:&lt;br&gt;• Multiple packet mode streams (minimum 4) (i.e. can access SlideShow / SPI data on extra sub-channel).&lt;br&gt;• Enhanced Packet Mode FEC protection</td>
<td>Packages/Classes/Interfaces:&lt;br&gt;org.universalradio.radio.*&lt;br&gt;org.universalradio.radioservice&lt;br&gt;org.universalradio.tuner&lt;br&gt;org.universalradio.radioservice.metadata&lt;br&gt;• Textual&lt;br&gt;• TextualDABDynamicLabel&lt;br&gt;• TextualMetadataListener&lt;br&gt;• Visual&lt;br&gt;• VisualDABSslideShow&lt;br&gt;• VisualMetadataListener&lt;br&gt;• TextualDABDynamicLabelPlusItem&lt;br&gt;• VisualIPRdnsRadioVis&lt;br&gt;•</td>
</tr>
</tbody>
</table>
## OMRI – Minimum Requirements

<table>
<thead>
<tr>
<th>Profile</th>
<th>General DAB requirements</th>
<th>Metadata/Data services (User applications)</th>
<th>API Classes</th>
</tr>
</thead>
</table>
| **Advanced receiver profile (optional in whole or part)** | Additional sub-channels to make the total simultaneous sub-channels 3 or more (e.g. to allow additional simultaneous decoding of other data services e.g. TPEG or Journaline) | User Applications:  
  - SPI (with delivery in MOT directory mode)  
  - Announcements  
  - Hybrid functionality  
    - SI, Logos  
    - PI  
    - RadioDNS  
    - Alternative Image  
  - Service Linking  
  - Additional character set decoding  
  - Other Ensemble functionality  
  - TII decoding | Packages/Classes/Interfaces:  
  org.universalradio.radioservice.metadata  
  - Group  
  - Location  
  - ProgrammeInformation  
  - ProgrammeServiceMetadataListener  
  - ServiceInformation  
  - SPIProgrammelnformation  
  - TermID |

DMB video service decoding
## LG Software stack

<table>
<thead>
<tr>
<th>Presentation to screen</th>
<th>Application</th>
<th>Functional processing</th>
<th>Functional processing</th>
<th>IP and other</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MOT, FIC/FIG processing</td>
<td>Service list, Basic and Advanced PAD features</td>
<td>Alternate image, logo updates</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LG Java API Converts C API to JAVA</th>
<th>Android Java API</th>
<th>Other C API</th>
<th>Other phone hardware</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>C API</th>
<th>Decodes and provides access to</th>
<th>Decodes and provides access to</th>
<th>Decodes and provides access to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controls</td>
<td>• MOT Data</td>
<td>• Parsed FIC for an Ensemble</td>
<td>• MOT datagroups</td>
</tr>
<tr>
<td>• Selecting Stations</td>
<td>• DLS Text</td>
<td>• Raw FIC</td>
<td></td>
</tr>
<tr>
<td>• Audio playout</td>
<td>• Signal quality data</td>
<td>• MOT datagroups</td>
<td></td>
</tr>
<tr>
<td>• Muting audio</td>
<td>• Event Callbacks, playing, bad signal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Tune to ensemble</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Radio chip hardware
- Receiver solution including hardware, driver and any software processing
- Other C API
- Other phone hardware
## OMRI-LG Software stack

<table>
<thead>
<tr>
<th>Presentation to screen</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional processing</td>
<td>Functional processing</td>
</tr>
<tr>
<td>MOT, FIC/FIG processing</td>
<td>Service list, Basic and Advanced PAD features</td>
</tr>
</tbody>
</table>

### OMRI API

- IRT Shim layer
- LG Java API Converts C API to JAVA
- Android Java API

### C API

- Controls
  - Selecting Stations
  - Audio playout
  - Muting audio
  - Tune to ensemble
- Decodes and provides access to
  - MOT Data
  - DLS Text
  - Signal quality data
  - Event Callbacks, playing, bad signal, good signal

### Other C API

- Decodes and provides access to
  - Parsed FIC for an Ensemble
  - Raw FIC
  - MOT datagroups

### Receiver solution including hardware, driver and any software processing

### Radio chip hardware

### Other phone hardware
## OMRI Software stack

<table>
<thead>
<tr>
<th>OMRI Java API</th>
<th>Android Java API</th>
<th>Other C API</th>
<th>Other phone hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>C API</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Controls</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Selecting Stations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Audio playout</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Muting audio</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Tune to ensemble</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decodes and provides access to</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• MOT Data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• DLS Text</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Signal quality data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Event Call backs, e.g. playing, good/bad signal</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decodes and provides access to</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Parsed FIC for an Ensemble</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Raw FIC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• MOT data groups</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• XPAD data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Packet mode data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• MSC data groups</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Receiver solution including hardware, driver and any software processing</td>
<td></td>
<td>Other C API</td>
<td>Other phone hardware</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Radio chip hardware</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TF OMRI next steps

• Next steps under the WorldDAB “OMRI” Task Force
  - TF chairman is Alex Erk, IRT
  - Establish Terms of Reference
  - Establish work programme
    ▪ OMRI API specification updates including functional enhancements
    ▪ API and example App code
    ▪ Standardisation routes
      • Android Developers website
      • ETSI
How the logos, social media and website asset links are updated using RadioDNS

**Project Logo demonstration**

Smartphone requests the web address of the selected station
(ECC,Eld,Sld,SCId)

User selects a station to listen to

RadioDNS returns the web address of the selected station
(FQDN / URL)

Smartphone contacts station website and requests the URL for RadioEPG

URL is returned to the smartphone

Smartphone requests SPI information via the returned URL

SPI info is returned to the smartphone for display and use.

Station website

RadioDNS.org
Advanced features demonstration

**Video to demonstrate**

- Click-through URLs
- Categorised Slideshow
- Logo based Service List
- Social media and web assets
Thank you