

HRADIO – HYBRID RADIO EVERYWHERE FOR EVERYONE

ALEXANDER ERK, MARKUS FRIEDRICH

"The absolute transformation of everything that we ever thought about music will take place within 10 years, and nothing is going to be able to stop it. Music itself is going to become like running water or electricity,"

David Bowie in der New York Times, 2002 ¹⁾

David Bowie war nicht nur als Künstler ein Visionär, sondern wenn man dieses New York Times-Interview liest und die Hintergründe zu David Bowies Engagement als ISP²⁾ (Internet Service Provider) und Betreiber eines SocialMedia Fan-Netzwerk „BowieNet“ kennt, wird man dieses Zitat und seine zukunftssträchtige Aussage noch besser einzuordnen wissen. 19 Jahre später ist ein Großteil dieser Vorhersage Wirklichkeit geworden. Der Zugriff auf Audioinhalte ist für heutige Hörer so selbstverständlich, wie den Wasserhahn aufzudrehen.

► David Bowie was not only a visionary as an artist, but if you read this New York Times interview and if you know the background to David Bowie's engagement as ISP²⁾ (internet service provider) and operator of a social media fan network called "BowieNet", you will understand this quote and his promising statement even better. 19 years later, a large part of this prediction has become reality. For today's listeners, access to audio content is as natural as turning on the water tap.

Der Erfolg smarter mobiler Geräte, der Mobilfunk und die Tatsache, dass Audioinhalte größtenteils „geräteneutral“ sind, bedeutet, dass Menschen fast alles, überall und zu jeder Zeit hören können. Zugleich ergeben sich mit der Explosion von neuen Audioangeboten als Podcasts und Streaming-Musikdienste mehr Möglichkeiten zur Auswahl als je zuvor.

Viele Zungen behaupten, dass diese Entwicklungen das Ende des linearen Rundfunks bedeuten und die Bedürfnisse der Hörer aus den nicht-linearen, personalisierten Streaming- und Downloadangeboten befriedigt werden. Ist das so?

Stellt man lineares live Radio und Streaming-/OnDemand Audio nebeneinander und vergisst für einen Moment den „Krieg der Welten...“, so birgt eine sinnvolle Vereinigung dieser beiden Formen ein enormes Potential.

Radio ist für viele seiner treuen Hörer ein Medium, das deshalb immer wieder eingeschaltet wird, weil es unterhält, informiert und von der Last des Aussuchens und selbst Aktivwerdens befreit. Es bietet Neues und Überraschendes sowie gleichzeitig eine Art „emotionale Heimat“ für den Hörer. OnDemand und Streaming dagegen bieten Kontrolle, (fast) grenzenlose Auswahl und ein direktes Feedback zu den Diensten.

Das hier vorgestellte EU-Förderprojekt HRADIO will genau an der Schnittstelle dieser beiden Welten den Radiosendern Mittel und Wege aufzeigen, wie solche hybriden Radiodienste aussehen und auf den gängigen Plattformen (Android, iOS und HTML/JS) realisiert werden können. Dabei hilft nur eine ganzheitliche Betrachtung der kompletten inhaltlichen und technischen Kette, um in allen Aspekten dem Hörer ein optimales Radioerlebnis zu bieten.

Technische Plattform

Radio wird zu häufig auf Empfangsweg und Audiodienste reduziert. Letztendlich will der Hörer seinen Lieblingssender genießen und interessiert sich wenig für die Tatsache, ob der gerade über UKW, DAB+ oder IP empfangen wird. Zuhause im WLAN ist IP in der Regel kein Problem, unterwegs im Auto sollte dann jedoch DAB+ genutzt werden. Wenn man aus dem regionalen Sendegebiet fährt, will man unter Umständen auch mobil über IP weiterhören, weil es gerade so interessant ist. Im Auto sind zusätzliche Funktionen wie zum Beispiel Service Following oder Alarm Announcements extrem wichtig. Bilder gehören in der heutigen Zeit zum Radio dazu und ein Achtjähriger, dessen Medienerlebnis von Streaming-Diensten wie zum Beispiel Spotify geprägt ist, fragt sich: „Wieso kann ich nicht einfach pausieren oder weiter skippen?“

Diese Anforderungen – vom „simplen linearen“ Dienst zu einem Mix aus verschiedenen Diensten und Nutzungen – lässt



Abbildung 1: Live Radio und On-Demand im Vergleich

1) <https://www.nytimes.com/2002/06/09/arts/david-bowie-21st-century-entrepreneur.html>

2) <https://arstechnica.com/information-technology/2016/01/david-bowies-isp-as-remembered-by-the-guy-who-helped-create-bowienet/>

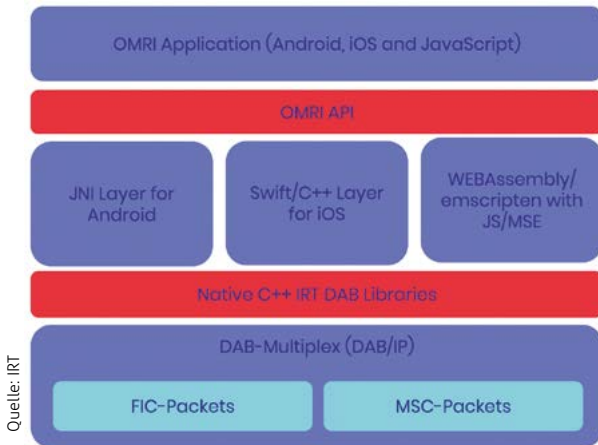


Abbildung 2: HRADIO-Applikationsarchitektur für iOS, Android und Web

die Komplexität in der Entwicklung von Radioapplikationen ansteigen. Damit diese Komplexität im HRADIO-Projekt auf die geplanten Funktionen reduziert und Applikationsentwickler von technischen Problemstellungen entlastet werden können, setzt das Projekt hauptsächlich auf die Nutzung von DAB+. Warum? Zuerst einmal ist DAB+ in nahezu allen Facetten und Zusatzdiensten standardisiert. Slideshow (inklusive Verlinkung), DynamicLabel+, ServiceFollowing, Alarm Announcements sind im DAB+ Multiplex definiert und werden „in Band“ synchron und optimal kodiert und übertragen. Kennt ein Applikationsentwickler DAB+ und hat er die Möglichkeit, DAB+ auf allen Plattformen (Car, iOS, Android und Web) zu nutzen, muss er nicht ständig im technischen Kontext umdenken und kann sich auf die Funktionalität konzentrieren. Um dies zu realisieren, wurde im Rahmen des HRADIO-Projekts folgende Architektur entwickelt:

Grundlage für die Entwickler ist die OMRI Spezifikation. OMRI (Open Mobile Radio Interface) ist eine API-Definition, die Applikationsentwicklern ermöglicht, Radiodienste, unabhängig von der darunterliegenden technischen Plattform (DAB+, UKW, IP, HD Radio...) zu nutzen. OMRI ist standardisiert durch WorldDAB+ (ETSI TS-103632 V1.1.1) und das HRADIO-Projekt hat sich zum Ziel gesetzt, OMRI für die wichtigsten Plattformen iOS, Android und WEB zu implementieren. Im Projekt wird hauptsächlich an der Android-Plattform gearbeitet, da Android auf mobilen Geräten die größte Nutzerbasis hat und durch Entwicklungen wie zum Beispiel Android Car und Android AutomotiveOS auch im Automobilsektor immer mehr Beachtung erfährt. Basis der OMRI-Implementierungen in HRADIO ist ein stabiles C++ 11 Framework für das Demultiplexing und die Dekodierung von DAB+ Ensembles. Für die Android-Plattform wird ein sogenannter JNI (Java Nativ Interface)-Layer implementiert, so dass die Android-Entwickler in den gewohnten Java/Kotlin-Umgebungen arbeiten können. In iOS sieht die Sache etwas einfacher aus, da C++ Quellen sehr einfach in iOS-Projekten genutzt werden können. Interessant wird die Sache auf Browser-Plattformen.

Webassembly (WASM) emscripten ist eine relativ neue Technik, um Webentwicklung auch mit anderen Sprachen jenseits von JavaScript zu ermöglichen. Nun ist gegen JavaScript generell erst einmal nichts einzuwenden, aber für viele Probleme ist es nicht unbedingt die erste Wahl. Darüber hinaus wird bei JavaScript der Sourcecode als Teil der Seite zum Browser übertragen und muss dort erst vor der Ausführung geparkt werden. Hier verspricht Webassembly

einen kleinen und schlanken Bytecode, der schnell geladen und ausgeführt werden kann. Erste unterstützte Sprachen sind C/C++, mittlerweile können aber auch Rust, C# oder Go genutzt werden. Die Browser von Mozilla, Google, und Apple unterstützen diese Technik und mit dem Compilertool „emscripten“ ist es relativ einfach möglich, eine existierende C/C++ Codebasis im Browser ausführbar zu machen. In unserer HRADIO-Web-Implementierung nutzen wir die gleiche C++ Codebasis wie für Android und iOS übersetzt mit emscripten im Browser. Sind die Ensembledaten erst einmal geladen und dekodiert, kann der Web-Entwickler in seiner JavaScript-Welt auch Slideshow, DynamicLabel und andere Techniken wie gewohnt nutzen. Die Audiodaten können an die MSE (Media Source Extensions) des Browsers wiedergegeben werden. Derzeit existieren noch Inkompatibilitäten bezüglich des Audioformats in DAB+, so dass zusätzlich zu den DAB+ Libraries auch ein AAC-Decoder, der den DAB+ AAC Stream (960 Frames/AU) unterstützt, per emscripten im Browser läuft.

DAB+-over-IP

Die Entscheidung, DAB+ als technische Grundlage für die HRADIO-Applikationsentwicklung zu nutzen, hat zur Folge, dass natürlich auch in allen Nutzungsszenarien ein DAB+ Signal zur Verfügung stehen muss. Generell unterstützt die OMRI-Implementierung die üblichen DAB+-USB-Empfangssticks. Dadurch steht grundsätzlich DAB+-Empfang zur Verfügung. Auf den mobilen Geräten jedoch stellen die externen Empfänger einen Fremdkörper dar und sind besonders mit der zum Empfang notwendigen Antenne recht unhandlich für den Nutzer. Aber auch an Orten, an denen kein DAB+-Signal zur Verfügung steht, soll – ganz im hybriden Gedanken des Projekts – auf IP-Empfang umgeschaltet werden können. Das HRADIO-Projekt hat sich daher entschieden, auch für den Fall, dass die Radiodienste über IP-Netzwerke übertragen werden müssen, auf DAB+ zu setzen und den Multiplex in IP-Streams zu übertragen. In der DAB+-Standardfamilie gibt es dafür den sogenannten EDI-Standard. EDI wurde ursprünglich für die Zuführung von DAB+-Ensembledaten vom Multiplexer zu den Sendestandorten konzipiert. Ein EDI-Ensemble enthält mehrere einzelne Radioservices und hat in der Regel eine Bandbreite von ca. 2,5 Mbit/s. Für den Unicast-Transport eines einzelnen Radiodienstes zum Client ist es natürlich nicht notwendig,

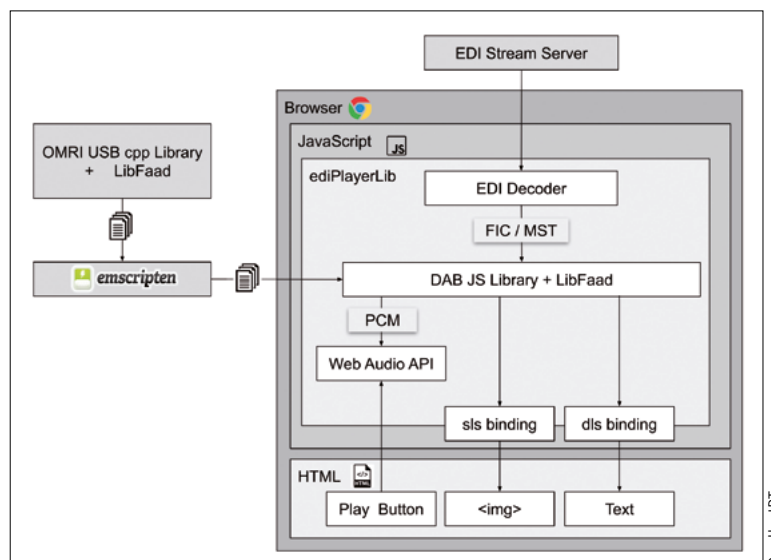


Abbildung 3: HRADIO-Playerarchitektur für Web/JavaScript

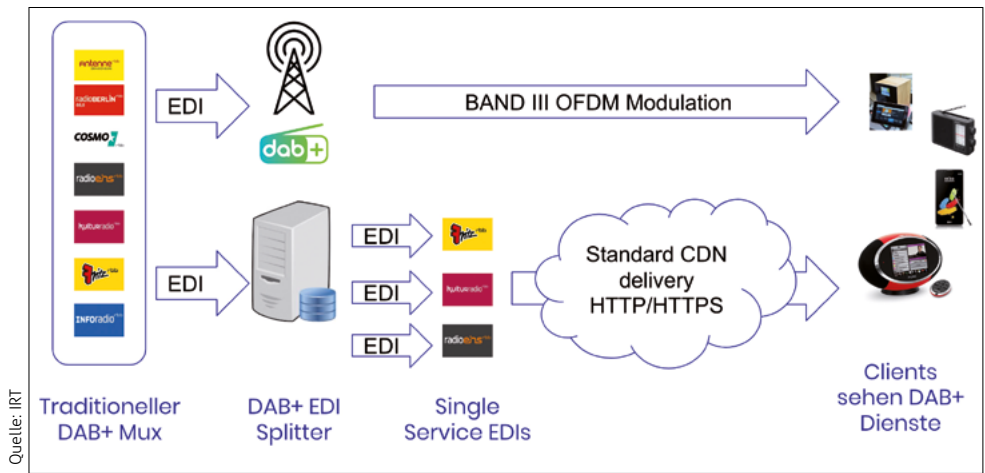


Abbildung 4: Übersicht des Signalflusses DAB+-over-IP

das ganze Ensemble zu übertragen. Daher wurde im Projekt eine Serversoftware implementiert, die von den DAB+-Multiplexern in den Rundfunkhäusern den kompletten EDI-Stream entgegennimmt und (entsprechend der Anzahl der enthaltenen Dienste) einzelne Single-Service-EDI-Streams für den IP-Transport zur Verfügung stellt.

Diese technische Lösung hat für alle Beteiligten große Vorteile: Die Applikationsentwickler arbeiten immer in der gleichen DAB+-Welt. Die Dienste, die genutzt werden (neben dem Audiosignal), sind „in Band“, synchron und deren Formate und Verhalten standardisiert. Ein Sender, der schon DAB+-Dienste für die terrestrische Übertragung generiert, kann diese ohne zusätzlichen Aufwand in den EDI-Splitter

schicken und als IP-Stream nutzen. Da für die Funktionen der HRADIO-Applikationen (siehe nächstes Kapitel) eine weitreichende Versorgung mit Metadaten und Signalisierungen notwendig sind, profitieren beide Plattformen (DAB+ und IP) von den Anstrengungen, die im Payout gemacht werden, gleichermaßen. Der EDI-Splitter ist in Google GO als sehr schlanker Serverdienst implementiert. Auf einem Raspberry Pi ist der Empfang, das Aufsplitten und die Weiterverteilung eines kompletten Ensembles kein Problem. Als nächster Schritt wird der EDI-Splitter um eine Time-Shift Funktion erweitert. Time-Shift von Radiodiensten ist ein zentraler Use-Case in den HRADIO-Applikationen. Derzeit ist die Timeshift-Funktion als lokaler Buffer auf den Android-Geräten realisiert,

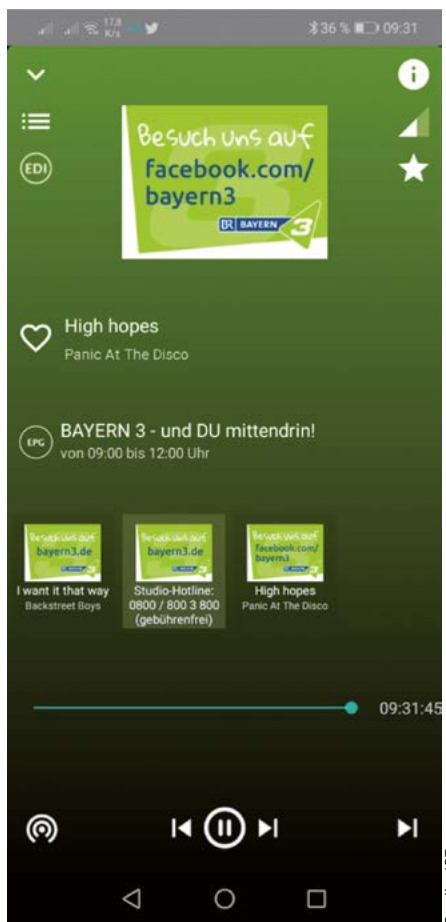


Abbildung 5: Bayern3 mit per DL+ segmentiertem Time-Shift buffer.

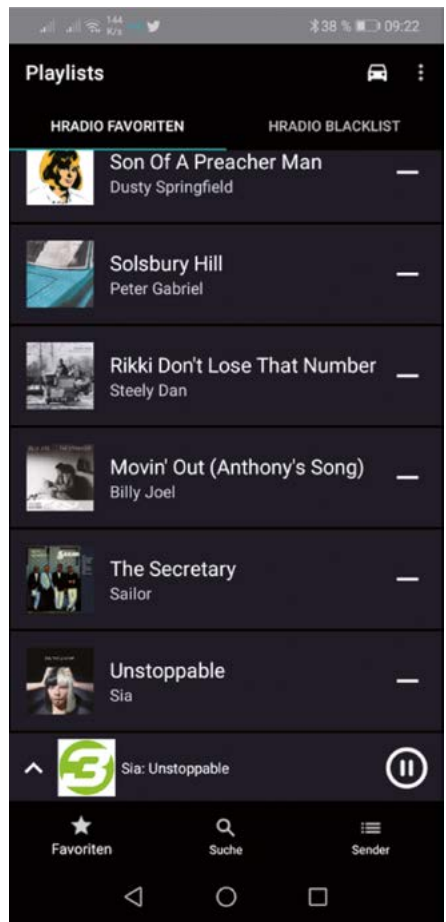


Abbildung 6: HRADIO-Favoriten-Playlist mit geliketen Songs aus dem laufenden Programm



Abbildung 7: „Im Interview der radioWelt“ Podcast

soll jedoch für die nächste Pilotphase des Projektes auf den Server verlagert werden. Dies hat einerseits den Vorteil, dass Clients entlastet werden. Andererseits steht sofort nach dem Starten des Services auf dem Client der gesamte Time-Shift-Buffer zur Verfügung und kann vom Hörer sofort vollumfänglich genutzt werden.

Neben den Bibliotheken für die Cliententwicklung liegt ein weiterer Fokus des HRADIO-Projektes auf der Entwicklung einer dezentralen und auf offenen Standards basierenden Plattform für die Bereitstellung von Service- und Programm-daten, die Sammlung von Nutzungsdaten (Feedback-Kanal) unter Berücksichtigung der Privatsphäre der Nutzer sowie die passgenaue, inhaltsbasierte Empfehlung von Radiosendern. Während es in einem klassischen Rundfunknetzwerk einfach möglich ist, die zur Verfügung stehenden Frequenzbänder abzusuchen (Frequenzscan), muss bei der internet-basierten Suche von IP-Diensten ein sogenannter Aggregator bereitgestellt werden, der Dienstinformationen sammelt und durchsuchbar macht.

Das HRADIO-Projekt nutzt bereits vorhandene RadioDNS-Dienstinformationen als Grundlage für eine verteilte Suchplattform, die für das Auffinden von IP-Diensten durch HRADIO-Clients konzipiert wurde. Die Plattform ist dabei in mehrere Module aufgeteilt: Das Importermodul sammelt RadioDNS SI- und SPI-Dateien, in IP-Streams eingebettete ICY-Tags sowie DynamicLabel+ Metadaten und reichert diese mit zusätzlichen Daten aus anderen Quellen an (zum Beispiel zusätzliche Informationen über Titel oder Interpret eines gespielten Lieds). Der so gewonnene Datensatz wird dann für die Durchsuchung optimiert im Datenbankmodul abgelegt und HRADIO-Clients über ein REST-API zur Verfügung gestellt.

Ein weiterer, wichtiger Aspekt hybrider Radiotechnologie ist die Sammlung von Hörerfeedback, das dann zur Verbesserung des Radioerlebnisses analysiert werden kann. Im Rahmen des HRADIO-Projektes wurde ein Plattformmodul entwickelt, welches Nutzungsdaten sammelt, dabei aber Rücksicht auf die Privatsphäre des einzelnen Benutzers nimmt. Der dahinterliegende Mechanismus basiert auf der aus der Psychologie bekannten Randomized-Response-Technik, wie sie auch in Googles Chrome Browser zur Sammlung von Telemetriedaten in stark erweiterter Form zum Einsatz kommt [1]. Dieses Verfahren bietet starke Privatsphäregarantien und erlaubt trotzdem eine sinnvolle Auswertung gesammelter Daten.

Neben dem Suchen und Finden von Radiosendern ist auch ein leistungsfähiger Empfehlungsdienst Teil der entwickelten Plattform. Um dabei weiterhin die Privatsphäre des Benutzers zu schützen, basieren Empfehlungen ausschließlich auf extrahierten Sendercharakteristika. Derartige Merkmale umfassen dabei zum Beispiel aus vorhandenen Metadaten gewonnene Genreverteilungen, per Speech-To-Text Algorithmen und Natural Language Processing (NLP) Verfahren extrahierte Schlüsselwörter sowie Reduktionen des Sender-Audiosignals auf komprimierte Fingerabdrücke mithilfe von Techniken des Maschinellen Lernens. Abbildung 8 zeigt eine 2D-Projektion der ermittelten Sendungs-Fingerabdrücke für eine Auswahl von 25 deutschen, öffentlich-rechtlichen Sendern. Gut zu erkennen ist hierbei die räumliche Nähe von Sendungen inhaltlich ähnlicher Sender.

Abbildung 9 liefert einen Überblick über die Architektur der Plattform. Alle Teildienste sind getrennt implementiert und werden in Docker-Containern verteilt und betrieben.

Der Clou des Systems ist, dass verschiedene Instanzen der Plattform zusammen in einer verteilten Föderation betrieben werden können und dabei den anfragenden HRA-

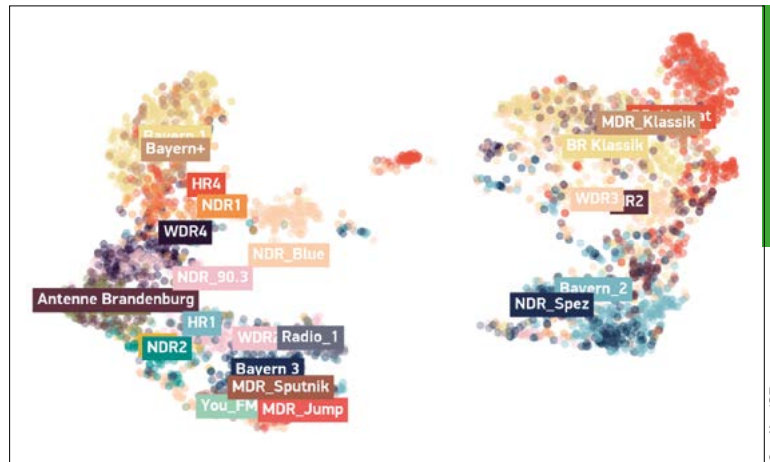


Abbildung 8 : Audio-Fingerabdrücke von insgesamt 5000 Sendungen verteilt auf 25 Sender.

DIO-Clients wie ein einzelnes System erscheinen. Hierzu pflegt jeweils eine Instanz ihren eigenen Suchindex mit den Adressen anderer Föderationsteilnehmer sortiert nach Kriterien, wie Ort oder Genreverteilung der dort gespeicherten Dienste. Bei der Bearbeitung einer Suchanfrage wird selbige dann an geeignete Plattforminstanzen weitergeleitet und deren Resultate zusammengeführt und zurückgegeben. Idee der Föderation ist es somit, ein verteiltes Netz aus Suchknoten aufzubauen, das Metadaten zu Radiosendern bereitstellt, ohne eine zentrale Instanz im Sinne eines Aggregators zu benötigen.

Dienste und Applikationen

Wie werden nun diese Funktionen der technischen Plattform in den Applikationen genutzt? Time-Shift ist ein zentraler Bestandteil der Nutzung. Sobald der Hörer einen Service auswählt und abspielt, wird das lineare Programm in einem Buffer aufgezeichnet und dieser Buffer entsprechend dem Ablauf im Programm segmentiert. Dafür werden die item.running- und item.toggle-Signale im DAB+ DL+-Datenstrom genutzt. Auf diese Art und Weise kann der TimeShift-Buffer entsprechend den Song- oder Beitragsgrenzen unterteilt werden und der Nutzer hat die Möglichkeiten, einzelne Lieder oder Beiträge (Nachrichten, Verkehr oder Wetter) direkt auszuwählen.

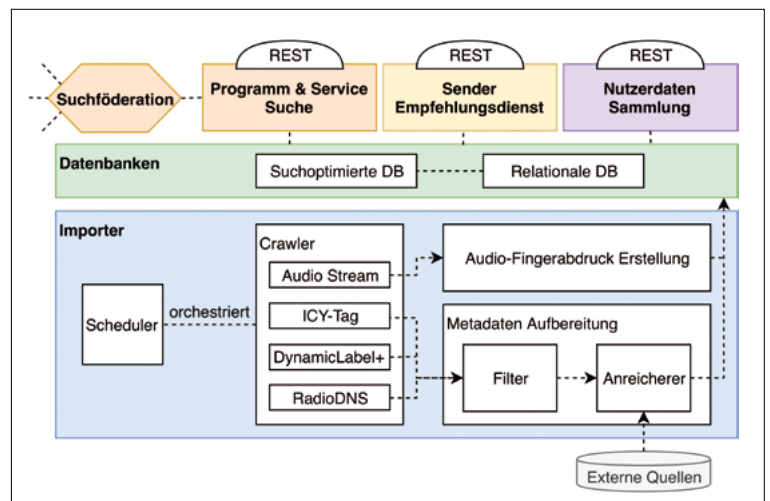


Abbildung 9: Architektur der entwickelten Plattform



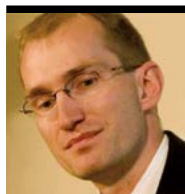
Abbildung 10: Übersicht HRADIO-Projektpartner

Abbildung 5 zeigt die HRADIO-App mit laufendem Radiodienst (Bayern3): Zwei Titel wurden anhand der DL+ Informationen im Ablauf erkannt und können direkt vom Nutzer angesprungen werden. Ähnlich wie bei Streamingdiensten kann per Skip-Tasten durch den TimeShift-Buffer navigiert werden.

Spotify und Podcasts

Die HRADIO-Applikation ist (wenn vorhanden) mit dem Spotify-Account des Nutzers verbunden. Titel, die über DL+ item.title und item.artist erkannt werden, kann der Nutzer liken und in einer Playlist auf dem Spotify-Konto speichern. Titel oder Beiträge (siehe Abbildung 6), die der Hörer nicht mag, kann er überspringen (skippen). Sind noch Sprungmarken im Time-Shift-Buffer vorhanden, werden diese angesprun-

Quelle: IRT



ALEXANDER ERK

leitet am Institut für Rundfunktechnik die Abteilung für Mediendienste und Anwendungen.

➤ www.irt.de

Quelle: Universität München



MARKUS FRIEDRICH

ist wissenschaftlicher Mitarbeiter am Lehrstuhl für Mobile und Verteilte Systeme an der LMU München.

➤ www.uni-muenchen.de

gen. Ansonsten wechselt der Player auf einen Titel aus dem Spotify-Account des Nutzers. In den Einstellungen können bevorzugte Playlists vom Nutzer konfiguriert werden.

Bietet ein Sender in den EPG-Daten Links auf programmbezogene Podcasts an, so können auch diese als Ersatzinhalte genutzt werden (siehe Abbildung 7).

Überspringt der Hörer auf diese Weise uninteressante Inhalte im laufenden Programm, so wird natürlich im Hintergrund weiterhin gepuffert und anhand der Item.toggle- und Item.running-Informationen geeignete Punkte für den Wiedereinstieg ins lineare Programm gesucht.

Mit der aktuellen Implementierung der App und des lokalen Time-Shifts bedarf es einer gewissen Zeit der Nutzung, bis der Buffer eine ausreichend große Anzahl von Beiträgen/Songs erkannt hat, um seinen vollen Nutzen demonstrieren zu können. Abhilfe soll hier der Umstieg auf eine serverbasierte Time-Shift Lösung bringen. Damit hat der Nutzer sofort ab dem Start eines Dienstes Zugriff auf alle Segmente des Puffers.

Fazit

Die acht HRADIO-Projektpartner (imec, IRT, LMU, W3C, VRT, Konsole, rbb und radioplayer) haben sich ein ehrgeiziges Ziel gesetzt: Ein auf allen Plattformen und Übertragungswegen einheitliches, personalisierbares und interaktives Radiosystem soll entwickelt und die komplette Kette demonstriert werden. Rundfunksender sollen alle dazu nötigen Signale einfach und standardisiert bereitstellen können (DAB+ und DAB+ over IP), Applikationsentwickler können sich voll und ganz auf die zu implementierende Funktionalität konzentrieren und einheitliche Schnittstellen auf allen Plattformen nutzen. Service- und Programmmetadaten werden dezentral und anonym aggregiert und bereitgestellt. Das Projekt läuft noch bis März 2020 und wird im Dezember 2019 eine letzte offene Pilotphase starten. Die Projektpartner erhoffen sich eine rege Beteiligung mit viel Feedback zu den demonstrierten Diensten und Funktionen.

Die Radiowelt steht vor großen Herausforderungen. Die Integration von linearen und nicht-linearen Diensten, die Versorgung der Angebote mit genauen, synchronen und aktuellen Metadaten, und eine gleichwertige Implementierung auf allen relevanten Plattformen ist essentiell für ein Funktionieren der angedachten Dienste und damit für den Erfolg bei den Hörern. Dies alles ist nur möglich, wenn vieles neu gedacht und umsichtig realisiert wird. Viel Arbeit liegt noch vor uns. Ein chinesisches Sprichwort sagt:

„Wenn der Wind der Veränderung weht, bauen die einen Mauern und die anderen Windmühlen.“

Die Digitalisierung weht auch dem guten alten Radio ins Gesicht. Deshalb müssen wir die Herausforderung annehmen und anfangen, Windmühlen zu bauen. ➤

<https://www.hradiio.eu/>

<https://www.irt.de/>

<https://www.uni-muenchen.de/>

Literatur

[1] Erlingsson, Ú., Pihur, V., & Korolova, A. (2014). RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In Proceedings of the ACM Conference on Computer and Communications Security (pp. 1054–1067). <https://doi.org/10.1145/2660267.2660348>