Technical White Paper

An Advanced Application Services Framework for Application and Service Developers using HD Radio™ Technology

Thom Linden

iBiquity Digital Corporation
101 Redwood Shores Parkway, Ste. 102
Redwood City, CA 94065

01 February 2003

The iBiquity Digital Corporation HD Radio™ system is designed to permit a smooth evolution from current analog Frequency Modulation (FM) radio to a fully digital In-Band On-Channel (IBOC) system. This system delivers digital audio and data services to mobile, portable, and fixed receivers from terrestrial transmitters in the existing AM and FM bands. Broadcasters may continue to transmit their existing analog AM and FM signal(s) simultaneously along with the new, higher-quality and more robust digital signals, allowing themselves and their listeners to convert from analog to digital radio while maintaining their current frequency allocation(s).

The HD Radio system allows multiple services to share the broadcast capacity of a single station. First generation (core) services include the Main Program Service (MPS) and the Station Information Service (SIS). This paper describes the concepts and foundation for an Advanced Application Services Framework which will provide next generation service and application developers with the application programming interfaces (API) required to rapidly develop, prototype and deploy their capabilities on future generation HD Radio platforms.

iBiquity Digital Corporation
8865 Stanford Boulevard
Suite 202
Columbia, MD 21045
410–872–1530 (Phone)
410–872–1531 (FAX)
info@ibiquity.com

Table of Contents

Introduction

The HD Radio system allows broadcasters to transmit their existing analog AM and FM signal(s) simultaneously along with new, higher-quality and more robust digital signals. The system also allows multiple services to share the digital broadcast capacity of a station within their current frequency allocation(s). First generation (core) services include the Main Program Service, reference [1] and [2], and the Station Information Service, reference [3]. With the digital capabilities of the HD Radio system, the broadcast industry has the opportunity to deliver additional, new digital information services to existing and new consumer audiences.

Examples of such services may include:

- On-demand audio-centric multimedia presentation of stock, news, weather and entertainment programming (See Figure 1 an advanced radio with audio, text and images)
- Broadcast updates to in-vehicle systems (e.g., maps for navigation systems)
- Services utilizing local storage (e.g. for time-shifted listening)
- Mobile commerce (e.g. with integrated cell phones)
- Targeted advertising (e.g. incentives to purchase the new CD)
- Reading services for the visually impaired
- Broadcast websites
- Traffic information (e.g. provided to navigation systems)

Figure 1 – Example of an on-demand multimedia radio

This paper describes the concepts and foundation for the Advanced Application Services (AAS) Framework. The framework provides a common infrastructure -- delineated by specific application programming interfaces (APIs) – to support a wide variety of new services. Availability of this infrastructure will allow a substantial degree of consistency and predictability, allowing next generation application and service developers to rapidly develop, test and deploy their capabilities on future generation HD Radio platforms.

What is a Service?

Einstein: "Everything should be made as simple as possible, but not simpler."

A service is simply the exchanging of information between a *Service Provider* and a *Service Consumer*. Service Providers can include broadcasters using their own station capacity as well as independent business entities simultaneously using (leasing) capacity from a broadcaster. Service Consumers are receiver-based applications that receive and process the information provided by a Service Provider. These applications may decode and render information from a service to a user (e.g., display text on the LCD of a radio or decode an audio stream) or simply pass the information on to a radio-attached device (e.g., software to update a navigation unit). In addition, a receive-side application can leverage multiple services of interest.

The information delivered within a service can contain objects that are discrete (e.g., a SMIL document, reference [4]) or can be continuous (e.g., streaming audio). In addition, receiver applications can uniquely distinguish services based on their *Service Class*, which is simply a means of categorizing services (e.g., On-demand audio versus real-time audio programs). Finally, services can be free or subscription based.

The proliferation of new services will increase not only the number of ways by which a broadcaster can reach listeners, but also the number and types of devices that can be reached. A broadcaster can simultaneously attract new 'listeners' (for audio-based services) and begin reaching other in-car and mobile devices such as:

- Navigation or Telematics Systems
- HD Radio enabled cell phones
- PDAs

What is the Advanced Application Services Framework?

The Advanced Application Services (AAS) Framework (see Figure 2) provides the 'bridge' or middleware between service providers and advanced HD Radio enabled receiver devices (service consumers).

The AAS Framework is comprised of four basic components:

1. The *Service Provider API (Service API)* – A common interface for service providers to access the bandwidth of one or multiple HD Radio enabled AM/FM stations.

2. *Ensemble Operations Center (EOC*) – Infrastructure subsystems that simultaneously accept distinct services for broadcast and support operational control over services based on contractual, bandwidth and security obligations. The EOC realizes the *Service API* for service providers.

3. *Advanced Application Platform (AAP)* – Embedded receiver subsystems that simultaneously receive distinct services, support audio decoding (including real-time PAC decoding), user interfaces, content storage, parsing and rendering. The AAP realizes the *Receiver Application API* for client receiver applications.

4. The *Receiver Application API (Application API)* – A common interface for applications to access the specific services of interest.

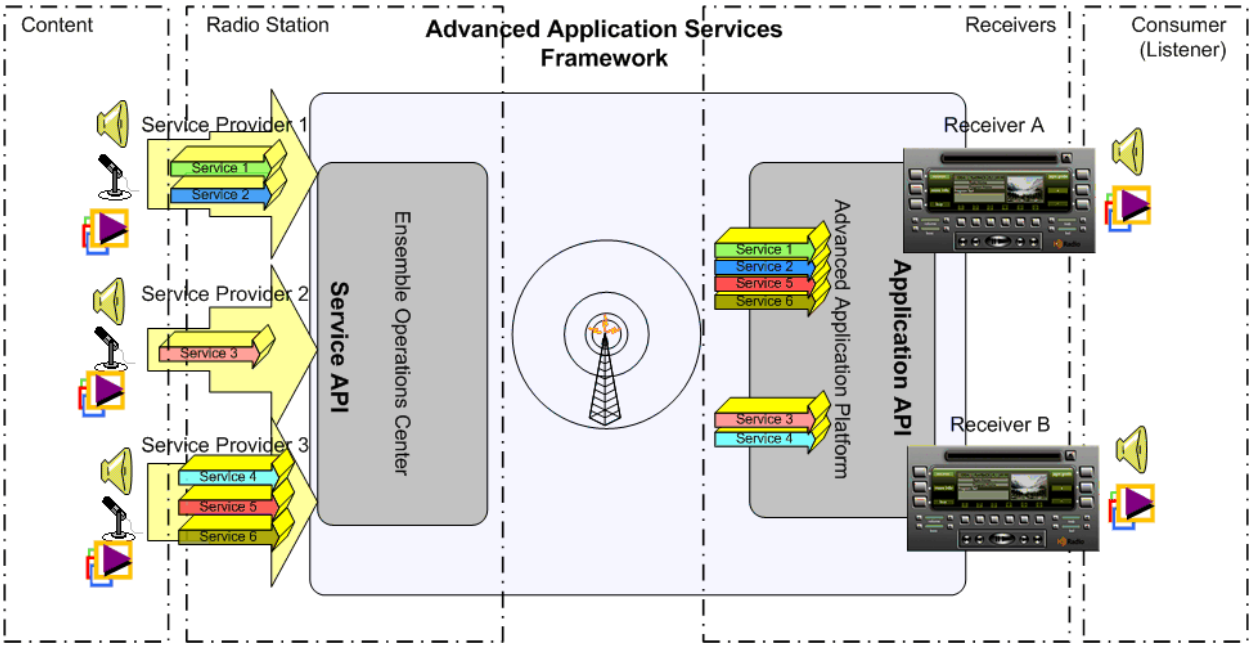

Figure 2 – AAS Framework provides the tools for connecting service providers with consumers

Who Benefits from this Framework?

The AAS Framework provides specific benefits in several areas of the broadcast chain:

- **Service Providers** are able to deliver content in a form that can be utilized by a wide variety of AAS-compliant devices. Also, a SDK 'software development toolkit' will be available to facilitate integration with a station's EOC.

- **Broadcasters** are able to effectively leverage their station's available data capacity using those services that best suit their business needs. These services can be originated local to a broadcast facility or by outside parties.

- **[Receiver] Application Developers** are able to leverage a powerful, modular AAS 'toolkit' (i.e., the Advanced Application Platform software libraries) resulting in rapid development of new products. In addition these developers can provide forward compatibility (for future services) and innovative solutions to their consumers.

- **End Users** are able to choose among a range of devices which can make use of new services offered by their favorite station(s). The AAS concept makes it possible for any number of receiver devices to interoperate with HD Radio enabled stations provi ding these services.

Where are Services Generated?

The AAS Framework may be viewed as a collaborative part of a larger networked environment (see Figure 3). This illustration shows Service Provider 1 (SP1) providing services to three broadcast stations, SP2 providing services to a single broadcast station, etc. It also illustrates that a broadcast station can originate services and distribute them to other nodes on the network. From this perspective, the AAS Framework provides a network bridge for distributing content over a HD Radio broadcast system.

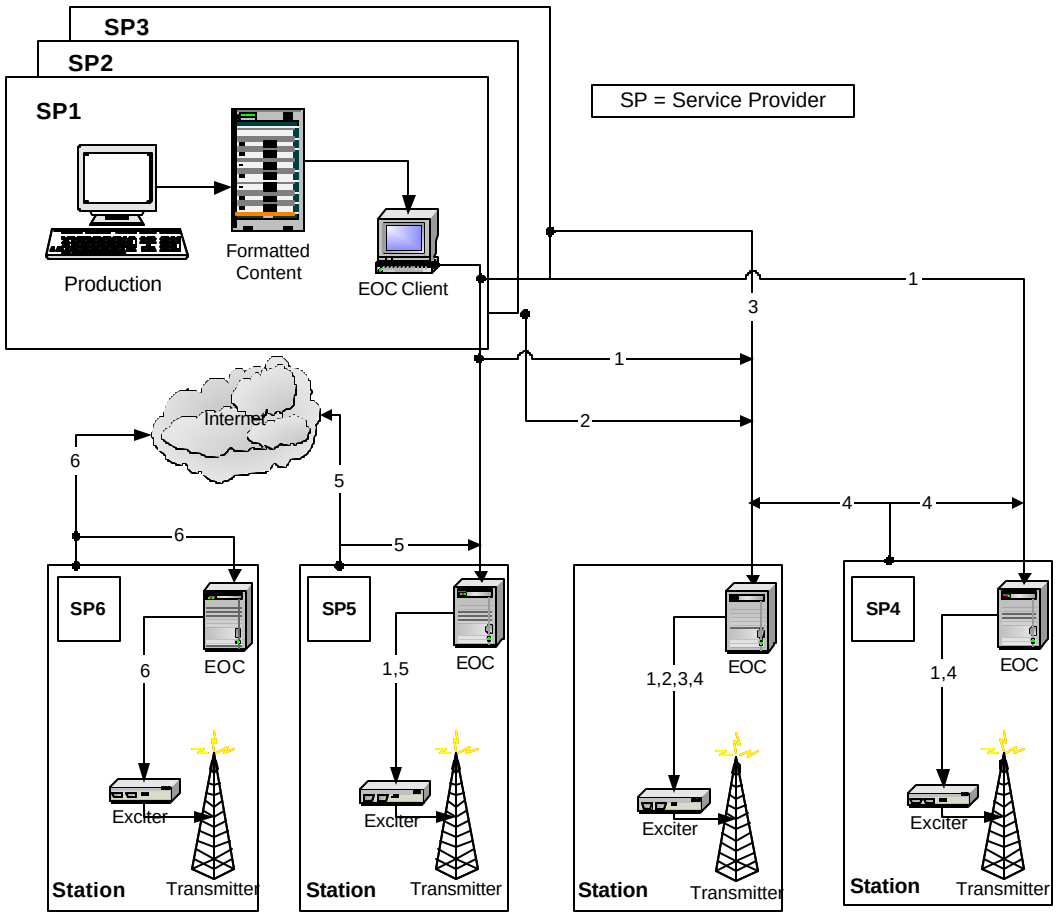

Figure 3 – Service Providers and Broadcasters can form powerful wide area networks to distribute their services to consumers.

AAS Framework Benefits

The AAS Framework provides implementation of specific abstractions that are available for Service and Application Developers. The goals are to:

- Ensure that the framework is portable to new architectures (e.g. the Microsoft® .NET architecture and new microprocessors), that it is fully international, and that it encapsulates environmental assumptions.
- Ensure that the development APIs are consistent, coherent, and clear.
- Encapsulate complexity to minimize risk and to allow for extension.
- Promote robustness by simplifying the coding effort required of developers.
- Provide transport transparency – the service doesn't need to know the underlying complexity of the transport or configuration of the transmission system. The client of a service (an application) doesn't need to know what kind of transport mechanism is used to move data from the service.

The AAS Framework also provides service management subsystems for Broadcast Operations. The goals are:

- Provide destination transparency – the same APIs control all communications, no matter what kind of service is involved.
- Provide consistent service registration and connection protocols for all new services to be added with minimum effort.
- Ensure operational metrics and controls are in place to enforce service bandwidth and time allocations.
- Ensure operational controls are in place to prevent security intrusions.

AAS Framework Functional Capabilities

The current AAS functional capabilities include[1]:

Broadcast Operations Functions

- Service Provider Registration, Deregistration
- Service Registration, Deregistration
    - Identification
    - Capacity Allocation(s), Allowed Connection Times
    - Quality of Service
    - Classification
- Service Activation, Deactivation
- Authentication
- Service to EOC Connection, Disconnection
- Service Announcement (SIS)
- Capacity Allocation, Modification
- Time-of-day (TOD) bandwidth/configuration scheduling
- Metrics
- Status Monitoring
- Usage Logging and Statistics

Service Developers Functions

- Service to EOC Connection, Disconnection
- Authentication
- Encryption
- Metrics
- Monitoring
- Streaming Broadcast
- Discrete Object Broadcast
- Error Reporting

Application Developers (Consumer Electronics) Functions

- Tuning (for real-time MPS and for AAS)
- Operating System Functions
- Client to Service Connection, Disconnection
- Authentication
- Conditional Access
- Decryption
- Metrics
- Monitoring
- Streaming Reception
- Discrete Object Reception
- Object Store and Retrieval
- Object Parsing
- Audio Recording and Playback
- Control Interface (i.e. interface to head-end micro controllers for user control functions)

[1] The AAS framework definition is ongoing.

Content Transmission in the AAS Framework

The Service and Application APIs support two broadcast communication models (see Figure 4): the discrete object model and the streaming model. The discrete model supports the transfer of finite-sized *objects*. In the latter, the service continuously transfers data with no absolute beginning, middle or end.

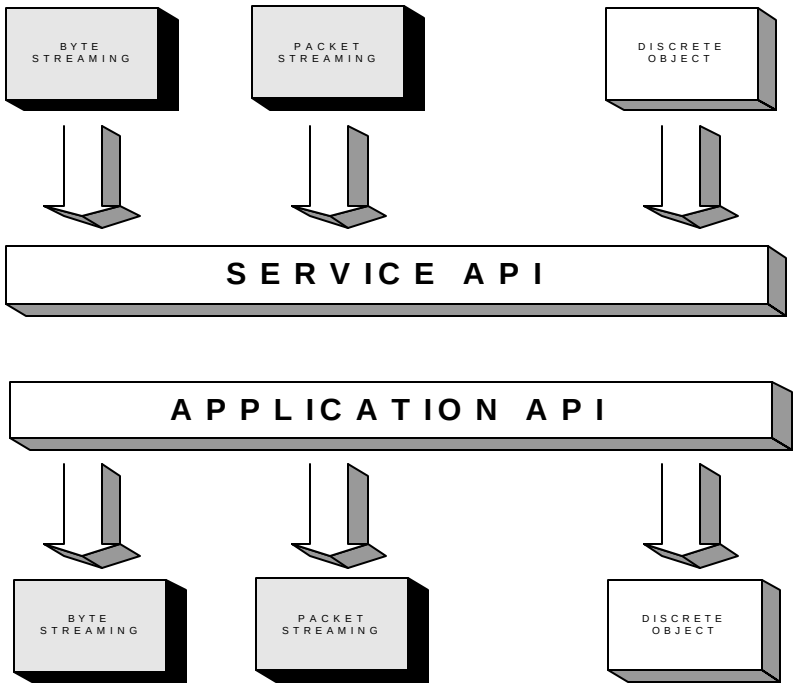

Figure 4 – Streaming and discrete broadcast communication models.

Services are classified in one of these communication models. At this juncture, it is unclear which will be the most commonly used, however the intent is to provide -- within the AAS Framework -- the foundation for a wide variety of services. A few examples of services enabled by these models are provided below[2]:

Byte Streaming

Stock Ticker Tape – a stream of character (text) information regarding stock prices such as "IBM up 2, Intel up 1 ……..".

TPEG (Transport Protocol Expert Group, reference [5]) – a stream of tokenized information relating to traffic and travel. TPEG implements its own framing and error recovery mechanisms.

Packet Streaming

Secondary "talk-entertainment" audio – a continuous stream of encoded audio packets where the client application is generally a low-bit rate decoder. Such a service might be suitable as a digital equivalent to existing analog reading services for the visually impaired.

[2] This is not intended to be a prediction of services which will be implemented, nor their success with consumers.

Discrete Object

Personal Radio – a service delivering on-demand audio-centric multimedia (see Figure 1 above) providing a collection of multimedia programming utilizing SMIL, reference [4], including stock, news, weather and entertainment.

Broadcast website – broadcasting all the files and the linking structure for a website. The client application would be a conventional HTML (hypertext markup language) browser.

Summary

HD Radio enables new services which can provide information in rich multimedia formats, tailored to the individual consumer interests and tastes. They represent new opportunities for additional revenue streams for Service Providers and Broadcasters.

The AAS Framework will provide a robust, flexible foundation for development of advanced services and applications. Developers will be able to rapidly prototype and deploy new services and new client applications. Broadcasters will be able to effectively manage the operational requirements to support multiple services and to efficiently manage bandwidth utilization.

This document represents work in progress. Revisions will be made available on the iBiquity web site http://www.ibiquity.com on a regular basis.

Acknowledgements

Several individuals at iBiquity contributed to this paper by providing numerous review comments. Thanks in particular to Mike West for his advice and commentary and to Dr. Serge Swerdlow for the initial diagrams.

Thom Linden, February 2003
Email: linden@ibiquity.com

Referenced Documents

[1] iBiquity Digital Corporation, “The Structure and Generation of Robust Waveforms for FM In-Band On-Channel Digital Broadcasting”, http://www.ibiquity.com

[2] iBiquity Digital Corporation, “HD Radio™ Air Interface Design Description – Main Program Service Data,” Doc. No. SY_IDD_1028s.

[3] iBiquity Digital Corporation, “HD Radio™ Air Interface Design Description – SIS Protocol,” Doc. No. SY_IDD_1020s.

[4] Kennedy, Tim, Mary Slowinski. *SMIL: Adding Multimedia to the Web*. Indianapolis, Indiana: SAMS, 2002.

[5] BBC TPEG Website, http://www.bbc.co.uk/rd/projects/tpeg/