Telecommun Syst
DOI 10.1007/s11235-010-9315-1

Standard audio format encapsulation (SAFE)

Homayoon Beigi · Judith A. Markowitz

© Springer Science+Business Media, LLC 2010

Abstract One characteristic that distinguishes speaker recognition (identification, verification, classification, tracking, etc.) from other biometrics is that it is designed to operate with devices and over channels that were created for other technologies and functions. That characteristic supports broad, inexpensive, and speedy deployments. The explosion of mobile devices has exacerbated the mismatch problem and the challenges for interoperability. This paper presents a detailed proposal for interoperability that supports all types of audio interchange operations while, at the same time, limiting the audio formats to a small set of widely-used, open standards. We call this proposal Standard Audio Format Encapsulation (SAFE). The SAFE proposal has been incorporated into speaker-recognition data interchange draft standards by the M1 (biometrics) committee of ANSI/INCITS and ISO/IEC JTC1/SC37 project 19794-13 (Voice data).

Keywords Speaker biometrics · Speaker verification · Speaker identification · Speaker recognition · Audio interchange · Audio encapsulation

1 Introduction

Unlike most other biometric modalities, speaker recognition (identification, verification, classification, tracking, etc.) is designed to operate over devices created for other technologies and functions. The ability to utilize the existing infrastructure supports deployment of broad, inexpensive and in an expeditious fashion. Such widespread deployment is counterbalanced by the variabilities produced by device and channel mismatch—issues that other biometrics are only beginning to face.

Interoperability is one of the principal benefits of, and rationales for, having standards. In research, lack of interoperability impedes the integration and use of data from disparate sources forcing researchers to devote precious time to achieving some kind of interoperability or to abandon valuable data. In private industry lack of interoperability is an impediment to system consolidation following mergers and acquisitions. Multi-factor authentication is beginning to dominate the security landscape but interoperability is a roadblock for organizations seeking to minimize errors through the use of speaker-recognition technology from multiple vendors. CentreLink, an Australian government agency providing unemployment benefits, designed such a system and found one of the most difficult and intractable problems to be audio data transfer among the engines [16]. Another source of interoperability problems is upgrades. The absence of interoperability impedes the use of data created by and for discontinued legacy systems. This problem gained heightened visibility when a leading vendor of speaker-verification technology released a new version of its engine that was not backward compatible. Use of a data-interchange standard that contains the proposal described herein (e.g., ANSI/INCITS draft standard INCITS 456 [1]) would ensure that the data already collected from user enrollment sessions for these and other kinds of systems could be reused.

Interoperability has become a challenge due to the explosion of mobile devices. Audio is a central modality of

H. Beigi (✉)
Recognition Technologies, Inc., 3616 Edgehill Road,
Yorktown Heights, NY 10598-1104, USA
e-mail: Beigi@RecognitionTechnologies.com

J.A. Markowitz
J. Markowitz Consultants, 5801 North Sheridan Road, Suite 19A,
Chicago, IL 60660, USA
e-mail: Judith@JMarkowitz.com

Published online: 26 May 2010

many of the new devices, in part because most of these devices can function as telephones but also because they are being asked to support the expanding use of speech recognition [13], speaker recognition [3], speech and voice search, audio indexing and multi-modal applications [18]. This rising dominance of mobile devices, the number of different codecs in those devices, the variety of audio formats involved, and the growth of speaker recognition applications have all exacerbated the device mismatched issue and have transformed the need for interoperability into a pressing security issue.

Despite the importance of interoperability for speaker-biometrics operations, there is remarkably little work on standardizing audio formats—even within the standards that govern the use of speaker recognition in those operations. The plethora of audio formats, the popularity of some proprietary formats, and the heterogeneity of the uses to which audio formats are being applied all give rise to the question of whether audio formats can be standardized.

The authors believe that standardization of audio formats is not only necessary but can be achieved by using the approach described in this paper. We call this proposal the Standard Audio Format Encapsulation (SAFE). It supports all types of audio-interchange operations while, at the same time, limiting the audio formats to a small set of widely-used, open standards.

2 Motivation

The initial idea of this proposal was to be able to use existing audio formats by bringing them under one cover so that different needs of the Speaker Biometrics community could be met without having to resort to using proprietary formats. Speaker recognition is not alone in this. Speech recognition, speech/voice search applications, and other audio processing systems all lack a comprehensive interoperability mechanism both within and across their specific domains. Interoperability in heterogeneous environments also constitutes a significant challenge for intelligent speech systems of all types.

For example, an increasing number of fusion-based systems are utilized to process the same audio clip for different purposes either to enhance the speaker recognition process or when speaker recognition is used as a component of a complex, intelligent system. These processes include language/dialect and style recognition (for multi-factor speaker identification or verification), audio segmentation (for finding structural sections), speech recognition (for content and context), event classification (for extraneous decisions), and music processing/search.

There are also cases where multiple speaker-recognition engines operate on the same audio clip. This approach is employed to increase accuracy, handle special cases, and achieve a spectrum of other task-specific objectives. Each of those engines may provide different kinds of analyses (e.g., text-dependent vs. text-independent) or they may use engines of the same type based on different algorithms as a means of enhancing accuracy. In some cases, those engines were developed by different vendors.

The number of applications that employ multiple speaker-recognition engines or multiple tools to process the same audio clip is growing. This trend is evident in both the private and public sectors.

Despite all of these trends, manufacturers seem to favor specific, proprietary formats. The reasons for such preferences vary, but often include a desire to escape the need to support the continually-expanding population of codecs and audio formats. Too often, these preferences lead to the establishment of proprietary technologies as de facto standards in a given segment of the market. That, in turn, produces a stranglehold on developers and manufacturers.

SAFE, the proposal in this paper, is designed to counteract the ascendancy of proprietary and IP-laden technologies. It not only supports multi-engine and multi-factor data exchanges—it promotes the spread of such applications by eliminating the need to support a plethora of formats, to contend with data incompatibilities, or to be held hostage by the owners of proprietary technologies.

In addition to interoperability, each step in the development of SAFE was guided by the following concerns:

- Are there any interchange requirements not covered?
- Are there any important features missing in general?
- Will any formats lose important features when converted?
- Any other compelling reasons to add more formats to the list?

3 Audio encapsulation standardization

Considering the various scenarios for the interchange of speaker-recognition data, three separate goals are most prevalent. Table 1 presents these conditions and the proposed audio format(s) for each case. This section describes the different cases in more detail.

Table 1 Audio interchange scenarios

Quality	Format
Lossless	Linear PCM (LPCM)
Amplitude Compression	μ-law (PCMU) and A-law (PCMA)
Aggressive variable bit-rate compression	OGG Vorbis
Streaming	OGG Media Stream

3.1 The uncompressed non-streaming case

Linear Pulse Code Modulation (LPCM) is the method of choice for this kind of audio representation [3]. There is no compression involved in either the amplitude domain or the frequency domain. The bare-minimum information needed in the header for this format is the number of channels, the sampling rate and the sample size (in bits). Table 2 includes this header data and some additional information. Microsoft WAV is not included because it is not a format; it is more of an encapsulation. WAV supports Linear PCM plus more than 104 other audio formats, most of which are proprietary coder-decoders (codecs) and many of which use some method of compression. Supporting WAV is tantamount to is supporting all the codecs which WAV supports. That violates the fundamental goal of interoperability of the SAFE proposal.

3.2 Amplitude compression with no streaming

Logarithmic PCM includes two algorithms which were proposed in the *G*.711 ITU-T (International Telecommunication Union-Telecommunication Standardization Sector) Recommendations of 1988 [10] operating at a sampling rate of 8-kHz with 8-bits per sample (64-kbps) with extensions to 80-kbps and 96-kbps as prescribed by the wide-band extension of *G*.711.1 [15].

In this scenario, the amplitude of the signal goes through some logarithmic transformation to increase the dynamic range of the signal. This conserves the number of bits needed to represent a sample. These two algorithms have been very effective techniques and have been used in telephony applications for many years. In the *G*.711 μ-law (PCMU) and *A*-law (PCMA) coding algorithms, each sample is coded to be represented by 8 bits with an 8-kHz sampling rate which amounts to a bit rate of 64 kbps. These two algorithms are known as PCMU and PCMA, respectively. Most telephony products use either PCMU or PCMA for capturing or recording audio. Supporting these algorithms should cover a wide variety of speaker-recognition applications.

Table 2 Audio format header

Type	Variable	Description
`U16`	ByteOrder	The value is `0xFF00` and it is set by the audio file producer.
`U16`	HeaderSize	Size of this header in bytes.
`Boolean`	Streaming	This will 0 for non-streaming and 1 for streaming. This boolean variable is redundant since the AF_FORMAT for streaming audio is greater than 0x0FFF. However, it is used for convenience.
`U16`	Compression	Standard Data Compression Scheme.
`U64`	FileLengthInBytes	In Bytes not including the header.
`U64`	FileLengthInSamples	In Number of samples.
`U16`	AudioFormat	See AF_FORMAT macros.
`U16`	NumberOfChannels	Number of channels. *N.B.*, Channel data alternates.
`U32`	SamplingRate	Sampling rate in samples per second. This is the audio sampling rate and not necessarily the sampling rate of the carrier which may be variable.
`U64`	AudioFullSecondsOf	It is the truncated number of seconds of audio.
`U32`	AudioRemainderSamples	This is the number of samples of audio in the remainder which was truncated by the above variable.
`U16`	BitsPerSample	Number of bits per sample, may be 0 for formats which use variable bits.

3.3 Variable bit-rate

These days, the first format that may come to mind is MP3. Unfortunately, MP3 is a proprietary format with many patents attached to it. In contrast, OGG Vorbis is an open-source, variable bit-rate format which, in most cases, performs as well as or better than MP3. Vorbis is the codec and OGG [8, 12] is the encapsulating mean for delivering the Vorbis codec [17]. There are also many open-source tools available including a library called LibAO which is available from the XIPH Open-Source Community for free [19].

3.4 The streaming case

The OGG media stream [8, 12], capable of streaming audio (and video) is included to cover the streaming case. It is completely open-source and can be used with many codecs including MP3. For the streaming case, though, it is recommended that only OGG Vorbis be used in conjunction with the OGG media stream.

3.5 Lossless compression

In most cases, speaker recognition applications may hesitate to utilize any amplitude or otherwise variable bit-rate lossy compression for fear of losing some features of the audio signal. For these cases, the linear PCM (LPCM) would be most suitable. However, the LPCM coding of the signal may produce very large audio files. To make this encapsulation standard more practical, several lossless compression schemes were considered. Again, the same considerations about using only open-source techniques led us to narrowing down the options down to three possible techniques. GnuZip of gzip, based on the Lempel-Ziv compression algorithm [20], has been known and widely used for general purpose compression for many years. Although this technique shows great performance, both in terms of size reduction and speed, it does have a flaw in its implementation and that is the lack of error-recovery. This means that if a single bit is displaced in a compressed file, the whole data is unusable. For this reason, although we have used gzip as a reference benchmark in Table 3, we do not recommend it for inclusion in this standard.

As an alternative, the bzip2 compression based on Burrows-Wheeler block sorting [4] and Huffman coding [9] is considered as an acceptable compression technique. Table 3 shows the performance of this technique with the default level 7 setting used in the compression and decompression of a 44-kHz audio file. The results of Table 3 have been reported in a normalized scale, where the size of the uncompressed audio file is considered to be 1.0, the coding time for the file using gzip is 1.0 and the decoding time for decompressing the compressed gzip file is also considered to be 1.0. All other numbers in the table are reported normalized according to the above reference values.

Table 3 Lossless compression [14] performance for a single channel 44-kHz LPCM file—The uncompressed file size is used as the reference unit (1.0) and the speeds of coding and decoding using gzip have been used as the reference units (1.0)

Compression method	Factor	Size	Coding time	Decoding time
gzip	1.4	0.7	1.0	1.0
bzip2	2.0	0.5	1.4	3.2
FLAC	2.2	0.46	0.7	1.7

Although bzip2 performs well, there is another compression technique which was developed by the Xiph group (the same group which is responsible for the development of OGG). This library is also freely available for most platforms from Xiph.org and is known as the Free Lossless Audio Codec (FLAC) [6]. This compression technique takes into account the fact that the compressed file is an audio file, so its performance exceeds that of the more general purpose compression libraries, gzip and bzip2. The performance of compression using FLAC has been reported in Table 3. In addition, [5] has done a comparison of most major lossless compression techniques with FLAC and has shown that it is the fastest lossless codec and its compression is within 3% of even the most complex codecs. It is important to note that the compression ratios of most practical lossless codecs is within 4% variability. Therefore, being the fastest such codec, being simple to implement, and even more importantly, being free convinced us to use it as the standard lossless compression codec in SAFE. It also seems like we are not alone in this conclusion [7] lists 30 home stereos, 2 automobile stereos, 23 Personal Digital Assistants (PDAs) and 5 other types of music players that support FLAC. 66 known artists and labels have adopted FLAC for their distribution codec and close to 70 audio software tools support FLAC [7].

4 Header

Table 2 contains the fields of the proposed data header. It (in conjunction with Tables 4 and 1) constitutes the core of this proposal. After the proposed header, the data format will follow, either as a whole or in the form of a stream which is handled by the OGG header immediately following the proposed header.

In a typical session there may be different *Instances* of audio which may have common information such as the sampling rate, the sample size, the number of channels, etc.

Table 4 Macros

Macro	Value
AF_FORMAT_UNKNOWN	0x0000
AF_FORMAT_LINEAR_PCM	0x0001
AF_FORMAT_MULAW	0x0002
AF_FORMAT_ALAW	0x0003
AF_FORMAT_OGG_VORBIS	0x0004
AF_FORMAT_OGG_STREAM	0x1000

Table 5 Compression type

Macro	Value
AF_COMP_NONE	0x0000
AF_COMP_BZIP2	0x0001
AF_COMP_FLAC	0x0002

This proposal assumes that any such feature will be set once as a default value and that it may be overridden later on, per instance, as the local instance information may change from the overall session information.

ByteOrder is a two-byte, binary code which is written at the time of the creation of the data. It is written as 0xFF00. When the data is read, if it is read as 0xFF00, it means that the machine reading the data has the same byte order as the machine writing the data. If it is read as 0x00FF, it means that the machine reading the data has a different byte order than the machine writing the data and that triggers a byte-swap which is applied to all subsequent information over one-byte in length.

Compression is a two-byte macro identifier which designates the lossless compression applied to the audio data. The possible values for this entry are given by Table 5.

FileLengthInSamples is a convenience measure for using LPCM, PCMU and PCMA. For these cases, *FileLengthInSamples* may be deduced from the *FileLengthInBytes*, *NumberOfChannels*, *SamplingRate* and *BitsPerSample*. It is not, however, readily computable for formats with a variable bit-rate compression. In order for it to be independent of the information which may be embedded in the encapsulated headers of OGG Vorbis, OGG Media Stream or any other format which may be added in the future, this value is included in the proposed header. Since *FileLengthInSamples* is designed for convenience, it may be set to 0.

AudioFullSecondsOf and *AudioRemainderSamples* define *FileLengthInSamples* when the number of samples is so large that an overflow may occur. *AudioFullSecondsOf* is the total number of seconds (in integer form) where the fractional remainder has been truncated. *AudioRemainderSamples* denotes the number of samples remaining in that truncated remainder. For example, if the total audio is 16.5 seconds long and if the sampling rate is 8-kHz, then *AudioFullSecondsOf* will be 16. The truncated remainder will then be 0.5 seconds which multiplied by 8000-Hz will produce 4000 samples which means the value of *AudioRemainderSamples* is 4000. This method of handling of the total number of seconds of audio avoids the use of floating point numbers which are most problematic in cross-platform interchanges. It also supports very long files where specifying the total number of samples can lead to an overflow.

5 Current status and future direction

The SAFE methodology was incorporated into the ANSI/INCITS 456 [1] draft standard which is currently undergoing a public review. ANSI/INCITS 456 is designed to be encoded in XML. Some of the vendors and integrators of speaker identification and verification wish to use SAFE in binary environments.

In order to facilitate this, we are working with Charles Johnson of the VoiceXML Forum's Speaker Biometrics Committee to develop a conversion tool. One version of the tool has already been created. Mr. Johnson is currently porting it to C++.

This work has also been recommended to the World Wide Web Consortium's Voice Browser Working Group (VBWG of the W3C) to be considered either as a complete standard or as a minimum requirement in the next version of its VoiceXML Standard (Version 3.0).

There are plans to include speaker identification and verification in that version of VoiceXML which, to date, has supported speech recognition, text-to-speech synthesis, and touch tone. We would like to see SAFE included in the planned speaker-recognition module.

SAFE has also been incorporated into the current working draft of "Biometric Data Interchange Formats—Part 13: Voice Data," designated as ISO/IEC Project 19794-13 [11].

6 Experimental results

In determining the suitability of the chosen codecs for speaker recognition, one must examine the effect of any lossy algorithms used in the process of storing and transmitting the audio. SAFE allows a combination of lossless and lossy codecs. It is apparent that lossless compression, by definition, does not modify the results of a speaker recognition system. Since the only lossy codec recommended in SAFE is the OGG Vorbis codec, its effects have been studied here to determine any degradation it may cause. In [2], an experiment in studying and treating time-lapse effects in speaker recognition was reported. In this study, the RecoMadeEasy®

Table 6 Acronyms and abbreviations

Acronym	Description
ANSI	American National Standards Institute
FLAC	Free Lossless Audio Codec
INCITS	InterNational Committee for Information Technology Standards
ISO	International Organization for Standardization
JTC	Joint ISO/IEC Technical Committee
SC	Subcommittee
SIV	Speaker Identification and Verification
VBWG	Voice Browser Working Group
WG	Workgroup
`U8,U16, U32,U64`	Unsigned 8, 16, 32 or 64-bit storage

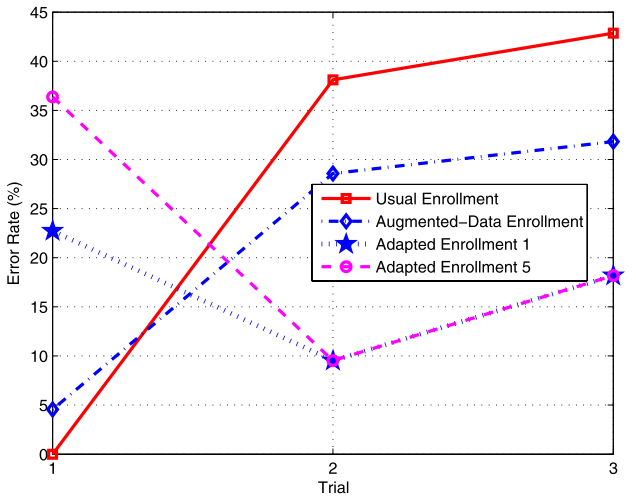


Fig. 1 Identification results using no adaptation, data augmentation and map adaptation

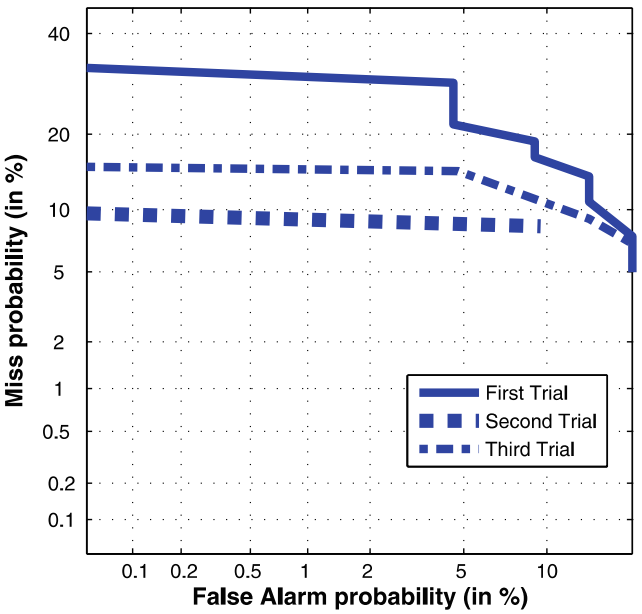


Fig. 2 Verification results using MAP adaptation

Speaker Recognition engine of Recognition Technologies, Inc. was used to report the effects of time-lapse on identification and verification results. Each of the 22 speakers in the report provided 3 recording sessions. The enrollment data was extracted from the first session. A supplemental amount of data from the first session was used as the base result and data from the second and third sessions were used for determining the effects of time-lapse. Figures 1 and 2 show the results of identification and verification that were reported in [2]. Here, to test the lossy effects of the Vorbis encoding, the same data that was used to produce Figs. 1 and 2 was first encoded using OGG Vorbis with a nominal bitrate of 128 kbps. Then, the audio was decoded back to Linear PCM and used to enroll, identify and verify the new audio files in the same manner as described in [2]. The results were identical. The same errors were repeated with the encoded/decoded audio files as with the original run reported in Figs. 1 and 2. This shows that the losses associated with the Vorbis codec are quite nominal and do not affect the results of speaker recognition significantly.

7 Conclusion

This paper has addressed the growing need for interoperability to support diverse speaker recognition applications running on heterogeneous devices and platforms and utilizing multi-vendor and multi-factor designs. We have demonstrated how a small set of well-established, and widely-used, open standards can be utilized to accomplish this goal. We call our implementation of this approach, SAFE.

The authors recognize that acceptance of SAFE or any proposal of this type may meet some resistance for reasons

that are more related to legacy than to technology. For example, companies that have made a practice of recording and storing audio using a variety of codecs supported by some incarnation of the WAV encapsulation or popular proprietary formats, such as MP3, may hesitate to accept the idea of converting their files to the standard formats supported by this proposal. This concern is a common source of resistance to standards of all types. We contend that our standards-based approach provides significant benefits, including enhancing the ability of developers and others to overcome changes in speaker-recognition technology and vendors.

Through the authors' efforts, SAFE is being incorporated into audio-interchange, draft standards for speaker recognition. It is a component of INCITS 456 [1] of the American National Standards Institute/International Committee for Information Technology Standards (ANSI/INCITS) which is in final INCITS management review. It is part of the standard being developed by the International Standards Organization/International Electrotechnical Commission Joint Technical Committee 1 Subcommittee 37 (ISO/IEC JTC1/SC37) Project 19794-13 (Voice Data) [11]. It has also been recommended to the Worldwide Web Consortium (W3C) Voice Browser Working Group for incorporation into Version 3 of its VoiceXML API standard.

References

1. ANSI/INCITS (2009). Project 1821—INCITS 456:200x, information technology—speaker recognition format for raw data interchange (SIVR-1). URL abstract: http://www.incits.org/abstracts/1821a.htm, purchase: http://www.techstreet.com.
2. Beigi, H. (2009). Effects of time lapse on speaker recognition results. In *16th internation conference on digital signal processing* (pp. 1–6).
3. Beigi, H. (2010). *Fundamentals of speaker recognition*. New York: Springer. ISBN: 978-0-387-77591-3.
4. Burrows, M., & Wheeler, D. J. (1994). *A block-sorting lossless data compression algorithm*. Tech. rep., Digital SRC Research Report.
5. Coalson, J. (2009). FLAC comparison.
6. Coalson, J. (2009). FLAC (free lossless audio codec).
7. Coalson, J. (2009). FLAC links.
8. Goncalves, I., Pfeiffer, S., & Montgomery, C. (2008). Ogg media types. RFC 5334 (proposed standard). URL http://www.ietf.org/rfc/rfc5334.txt.
9. Huffman, D. (1952). A method for the construction of minimum-redundancy codes. *Proceedings of the Institute of Radio Engineers*, *40*(9), 1098–1101.
10. ITU-T (1988). G.711 pulse code modulation (PCM) of voice frequencies. ITU-T recommendation. URL http://www.itu.int/rec/T-REC-G.711-198811-I/en.
11. JTC1/SC37, I. (2009). Text of 3rd WD 19794-13 biometric data interchange formats—part 13: voice data. URL http://isotc.iso.org/livelink/livelink/JTC001-SC37-N-3053.pdf?func=doc.Fetch&nodeId=7941680&docTitle=JTC001-SC37-N-3053.
12. Pfeiffer, S. (2003). The ogg encapsulation format version 0. RFC 3533 (informational). URL http://www.ietf.org/rfc/rfc3533.txt.
13. Rabiner, L., & Juang, B. H. (1993). *Fundamentals of speech recognition*. *Prentice Hall signal processing series*. New York: Prentice Hall. ISBN: 0-13-015157-2.
14. Salomon, D. (2006). *Data compression: the complete reference* (4th ed.). New York: Springer. ISBN: 1-84-628602-6.
15. Sollaud, A. (2008). RTP payload format for ITU-T recommendation G.711.1. RFC 5391 (proposed standard). URL http://www.ietf.org/rfc/rfc5391.txt.
16. Summerfield, R., Dunstone, T., & Summerfield, C. (2008). Speaker verification in a multi-vendor environment. In *W3C workshop on speaker identification and verification (SIV)*.
17. *0.8* 1.2 Vorbis I Specifications (2004). The XIPH open-source community. URL http://xiph.org/ao/doc/.
18. Viswanathan, M., Beigi, H. S., Dharanipragada, S., Maali, F., & Tritschler, A. (2000). Multimedia document retrieval using speech and speaker recognition. *International Journal on Document Analysis and Recognition*, *2*(4), 147–162. Invited paper.
19. Libao ogg audio api. (2004). The XIPH open-source community. URL http://xiph.org/ao/doc/.
20. Ziv, J., & Lempel, A. (1977). A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, *23*(3), 337–343.



Homayoon Beigi earned his Bachelors, Masters and Doctorate degrees from the School of Engineering and Applied Sciences of Columbia University in 1984, 1985 and 1991 respectively. He holds three employment positions. For the past few years, he has been involved in research and development in the fields of Biometrics, Pattern Recognition and Internet-Commerce. As the President of Recognition Technologies, Inc., he has been conducting research toward the production of a series of Speaker Recognition, Language Modeling and Signature Recognition line of software engines called RecoMadeEasy (TM). His work as the Vice President of Internet Server Connections, Inc. includes the development of the multiple-award winning CommerceMadeEasy(R) software, an elaborate Electronic Commerce system used world-wide. In addition, as an Adjunct Professor of Mechanical Engineering, he has taught "Applied Signal Recognition and Classification," "Speech and Handwriting Recognition," and "Digital Control" at the mechanical and electrical engineering departments of Columbia University. He was a Research Staff Member at the IBM T.J. Watson Research Center from 1991 to 2001 where he conducted research on Speaker Recognition, Language Modeling, Aggressive Search Techniques, Speech Recognition, On-Line Handwriting Recognition, Software Architecture, Control Theory and Neural Network Learning. Homayoon Beigi is actively involved in several standards bodies. He is an active liaison and the driving force behind the Audio Format for ANSI/INCITES M1 committee. He is also an active liaison in the US delegation of the ISO/SC37-JTC1-WG3. In addition, he is an active liaison in the VoiceXML Forum committee on Speaker Biometrics. His other research activities include Image Compression, Kinematics, Dynamics, Financial Optimization, and Zero-Gravity Fluid Dynamics. He has many publications including the first textbook ever written on Speaker Recognition, entitled, "Fundamentals of Speaker Recognition."



Judith A. Markowitz is president of J. Markowitz, Consultants, a strategic consulting firm specializing in speech processing and voice biometrics. She is recognized internationally as a leading analyst in voice biometrics and as a thought leader in speech processing. She is also known for her work on lexical-semantic networks. In 2003, she was named a top ten leader in speech processing and in 2006 she was elevated to Senior Member of the IEEE.

Judith co-chairs the VoiceXML Forum's Speaker Biometrics Committee. She's the Forum's liaison to ANSI/INCITS M1 (biometrics) and ISO JTC1/SC37 (biometrics). She's editor of INCITS 456 "Speaker Recognition Format for Raw Data Interchange," which is now in public review. She co-founded and co-chaired the Speaker Verification API Standard Workgroup (SVAPI) and participated in the development of BioAPI, ANSI X9F4 (management and security of biometrics in financial services), and ANSI/HFES 200 (human-factors for speech).

Judith's publications include the book Using Speech Recognition, book chapters (e.g., Designing for Speaker Authentication. In D. Dahl (Ed.), Practical Spoken Dialog Systems and two articles in the S.Z. Li (Ed.), Encyclopedia of Biometrics), journal articles (e.g., "Voice Biometrics: Speaker Recognition in the Real World" in Communications of the ACM), and the Voice ID reports on speaker biometrics. She currently writes monthly reports monitoring the voice-biometrics industry.

Judith has a master of science degree (computer science, artificial intelligence focus) from DePaul University and a doctoral degree (linguistics) from Northwestern University.