Voice Format for Data Interchange (VRR10)

Source: Project Editors

Date: November 27, 2006

Revision History

Revision	Date	M1 Document	Comments
1	27-Nov-2006		Speaker Biometrics Format for Data Interchange Internal Working Draft version 0.3

Co-Editor Contact Information: Dr. Judith Markowitz, VoiceXML Forum
email: Judith@JMarkowitz.com
phone: (773) 769 9243
fax: (773) 769 9253

Co-Editor: Guy Cardwell, Motorola
Co-Editor: Artour Karaguiozian, Motorola

Abstract This work is the product of collaboration between the Speaker Biometrics Committee of the VoiceXML Forum (SBC), a liaison member of M1, and ANSI/INCITS/M1 (biometrics). It defines a method for characterizing the speech produced by an end user for enrollment, verification, or identification that is predicated on the concept of a dialog and turns within the dialog. It supports transmission of raw speech data with an optional extension for proprietary data. It defines the attributes that are needed to generate a voice model from the dialog and turns and includes the XML representation of those attributes.

(this page intentionally left blank)

Table of Contents

Table of Figures

Tables

Foreword

This work is the product of collaboration between the Speaker Biometrics Committee of the VoiceXML Forum (SBC), a liaison member of M1, and ANSI/INCITS/M1 (biometrics). It defines a method for characterizing the speech produced by an end user for enrollment, verification, or identification that is predicated on the concept of a dialog and turns within the dialog. It supports transmission of raw speech data with an optional extension for proprietary data. It defines the attributes that are needed to generate a voice model from the dialog and turns and includes the XML representation of those attributes.

This document contains one annex that is normative (Annex A), one annex that is informative and is not considered part of the standard (Annex B), and a bibliography (Section 7) that is not considered part of the standard.

The Voice Extensible Markup Language Forum (VoiceXML Forum) is a standards body serving the speech-processing industry. It was formed in 1999 by AT&T, IBM, Lucent, and Motorola with a mission to establish a standard language for speech-processing technologies that would support interaction between telephone and Internet channels. It released VoiceXML version 1.0 in 1999 and it established a partnership with the W3C in 2000 to co-develop standards for speech-processing technologies that would enable them to operate on the Internet and interoperate with other Internet standards. Since the formation of the partnership the Forum's role has shifted from developing standard specifications to

- Developing and implement VoiceXML certification programs
- Marketing and education related to VoiceXML
- Identifying new directions for VoiceXML (e.g., adding speaker biometrics to the VoiceXML language)
- Developing requirements related to incorporating those new directions into VoiceXML

Today, the Forum has almost 400 members and, with over 10,000 deployments, the VoiceXML language has become the core standard within the speech-processing industry.

INCITS (The International Committee for Information Technology Standards) is the ANSI recognized Standards Development Organization for information technology within the United States of America. Members of INCITS are drawn from Government, Corporations, Academia and other organizations with a material interest in the work of INCITS and its Technical Committees. INCITS does not restrict membership and attracts participants in its technical work from 13 different countries, and operates under the rules of the American National Standards Institute.

In the field of Biometrics, INCITS has established the Technical Committee M1. Standards developed by this Technical Committee have reached consensus throughout the development process and have been thoroughly reviewed through several Public Review processes. In addition, this American National Standard has been approved by the INCITS Executive Board and ANSI Board of Standards Review for Publication as an ANSI/INCITS Standard.

Technical Committee M1, Biometrics, which reviewed this standard, had the following members:

Fernando Podio, Chair

Wayne Kyle, Vice-Chair

[table of M1 members goes here]

Task Group M1.3 on Biometric Data Interchange Formats, which developed this standard, had the following members:

name goes here], Chair and

name goes here] , Vice-Chair

[table of M1.3 members goes here]

At the time this proposal was presented to M1 and the VoiceXML Forum membership, the Speaker Biometrics Committee of the VoiceXML Forum, which developed this proposal, had the following members

Judith Markowitz, Co-chair

Ken Rehor, Co-chair

[table of VoiceXML Speaker Biometrics Committee members goes here]

This proposal was developed with the support of the following vendors, integrators, and consultants who are not currently active members of the Speaker Biometrics Committee

[table of vendors and consultants goes here]

Introduction

This ANSI/INCITS standard defines a method of representing SIV information using the concept of dialog and turns within a dialog. A dialog is the verbal interaction between an end-user and computer or another human during the length of an SIV session. By describing the audio transmission channel, the speech, and other information, a dialog can be parsed and the end-user's voice analyzed in an efficient and interoperable manner. Algorithms that are compliant with this standard must observe the format and XML syntax described. This includes: order and size of fields / elements, presence of all required elements, adherence to range limits on values, and internal consistency (the number of single turn records must match the number of turns specified as having been taken in the dialog, for example).

Standards that are sources for the work herein include the ETSI MRCP v2 ([MRCP]), ANSI/INCITS 358-2002 ([BioAPI1)] along with the international version of BioAPI ISO 19784-1 ([BioAPI2]), NISTIR 6529-A-2003 (the Common Biometric Exchange Framework Format [CBEFF1]) as well as more recent versions of CBEFF ([CBEFF2], [CBEFF3]).

The data record specified in this standard will be embedded in a CBEFF-compliant structure in the CBEFF Biometric Data Block (BDB) or will be used within a VoiceXML 3.X . The representation will be in the W3C's XML 1.0 language ([XML]).

1 Scope

This standard specifies a concept and data format for representation of the human voice at the raw-data level with optional inclusion of non-standardized extended data. The data format is generic in that it may be applied to and used in a wide range of application areas where automated and human-to-human SIV is performed. No application-specific requirements, equipment, or features are addressed in this standard.

The standard contains definitions of relevant terms, a description of the basic verbal interaction called a "dialog," a data format for containing the data, and conformance information.

2 Conformance

A system conforms to this standard if it satisfies the mandatory requirements herein for representing the information in a speaker-recognition dialog as described in Sections 5 and 6 and Annex A.

3 Normative References

The following standards contain provisions that, through reference in this text, constitute provisions of this standard.

ANSI/INCITS 358 [BioAPI1]
ANSI/INCITS (American National Standards Institute/The International Committee for Information Technology Standards). *The BioAPI Specification.* Washington, DC: American National Standards Institute, 2002.

ANSI/INCITS 398 [CBEFF2]
ANSI/INCITS (American National Standards Institute/The International Committee for Information Technology Standards). *Common Biometric Exchange Formats Framework (CBEFF).* Washington, DC: American National Standards Institute, 2005.

IEEE 754 [IEEE-754]
IEEE (Institute of Electrical and Electronics Engineers). *Standard for Binary Floating-Point Arithmetic.* New York, NY: Institute of Electrical and Electronics Engineers, Inc.,1985.

IETF MRCP [MRCP]
IETF (The Internet Engineering Task Force). *Media Resources Control Protocol version 2.0 draft-ietf-speechsc-mrcpv2-11*. The Internet Engineering Task Force, 2006. (see http://www.ietf.org/internet-drafts/draft-ietf-speechsc-mrcpv2-11.txt）.

IETF RFC [RFC]
IETF (The Internet Engineering Task Force). Alvestrand, H. (Ed.). *Tags for the Identification of Languages.* The Internet Engineering Task Force, 2001. (see http://www.ietf.org/rfc/rfc3066.txt).

ETSI GSM 05.03 [GSM]
ETSI (European Telecommunication Standards Institute). *Digital cellular telecommunications system; Channel Coding* ETSI Technical Specification. Sophia Antipolis, France: European Telecommunication Standards Institute,1999.

ISO 639 [LANG1]
ISO (International Standards Organization). *Codes for the representation of names of languages*. Geneva: International Standards Organization, 2002.

ISO 639-6 [LANG2]
ISO (International Standards Organization). *Codes for the representation of names of languages -- Part 6:*

Alpha-4 representation for comprehensive coverage of language variation. (under development) Geneva: International Standards Organization, 2006.

ISO/IEC 846 [ASCII]
(International Standards Organization/ International Electrotechnical Commission). *ISO 7-bit coded character set for information interchange.* Geneva: International Standards Organization, 1991.

ISO 10646 [ISO10646-1]
ISO (International Standards Organization/ International Electrotechnical Commission). *Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*; Geneva: International Standards Organization, 2000.

ISO 10646 [ISO10646-2]
ISO (International Standards Organization/ International Electrotechnical Commission). *Universal Multiple-Octet Coded Character Set (UCS) – Part 2:Supplementary Planes.* Geneva: International Standards Organization, 2001 (and updates).

ISO 19784-1 [BIOAPI2]
ISO (International Standards Organization/ International Electrotechnical Commission). *Text of ISO/IEC FDIS 19784-1, Information technology – Biometric application programming interface – Part 1: BioAPI specification.* Geneva: International Standards Organization, 2005.

ISO 19785 [CBEFF3]
ISO (International Standards Organization/ International Electrotechnical Commission). *Common Biometric Exchange Formats Framework -- Part 1: Data element specification.* Geneva: International Standards Organization, 2006.

ISO 3166 [CTRY2]
ISO (International Standards Organization). *Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes.* Geneva: International Standards Organization, 1997.

ISO 8879 [ISO8879]
ISO (International Standards Organization). *Standard Generalized Markup Language (SGML)*. Geneva: International Standards Organization, 1986.

ISO UTF-8 [UTF-8]
ISO (International Standards Organization/ International Electrotechnical Commission). *Universal Multiple-Octet Coded Character Set (UCS) -- Part 1: Architecture and Basic Multilingual Plane 10646-1:2000 Annex D*. Geneva: International Standards Organization, 2000

ITU-T ADPCM [ADPCM]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)*. Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1990.

ITU-T AMR [AMR]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB).*
Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 2003.

ITU-T G.711 [G711]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Recommendation G.711* - (STD.ITU-T RECMN G.711-ENGL). Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1989.

ITU-T G.722 [G722]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *7 kHz audio-coding within 64 kbit/s.* Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1988.

ITU-T G.723 [G723]
(International Telecommunication Union –Telecommunication Standardization Sector). *Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s* (to be published).
Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 2006.

ITU-T G.728 [G728]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Coding of speech at 16 kbit/s using low-delay code excited linear prediction.* Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1992.

ITU-T G.729 CS-ACELP [CELP]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector).
G.729: Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP). Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1996.

ITU-T G.729A [G729A]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Reduced complexity 8 kbit/s CS-ACELP speech* codec (Annex A). Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1996.

ITU-T G.729B [G729B]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *A silence compression scheme for G.729 optimized for terminals conforming to Recommendation V.70* (Annex B). .
Geneva: International Telecommunication Union –Telecommunication Standardization Sector, 1996.

ITU-T P56 [P56]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Objective measurement of active speech level.* . Geneva: International Telecommunication Union – Telecommunication Standardization Sector, 1993.

ITU-T PCM [PCM]
ITU-T (International Telecommunication Union –Telecommunication Standardization Sector). *Pulse code modulation (PCM) of voice frequencies*. Geneva: International Telecommunication Union – Telecommunication Standardization Sector, 1988.

NIST IR 6629-A [CBEFF1]
NIST (National Institute of Standards in Technology). *CBEFF: Common Biometric Exchange Formats Framework*. Gaithersburg, MD: National Institute of Standards in Technology, 2003.

Unicode [Unicode]
The Unicode Consortium. *The Unicode Standard, Version 2.0.* Reading, MA: Addison-Wesley Developers Press, 1996.

Unicode3 [Unicode]
The Unicode Consortium. *The Unicode Standard, Version 3.2.* defined by: *The Unicode Standard, Version 3.0.* Reading, MA: Addison-Wesley Developers Press, 2000. as amended by the *Unicode Standard Annex* #27: *Unicode 3.1* (http://www.unicode.org/reports/tr27/) and *Unicode Standard Annex* #28: *Unicode 3.2* (http://www.unicode.org/reports/tr28/).

VoiceXML2 [VXML]
W3C (World Wide Web Consortium). *Voice Extensible Markup Language (VoiceXML) Version 2.1.* World Wide Web Consortium, 2004. (see http://www.w3c.org/TR/2004/WD-voicexml21--20040728/)

W3C XML Schema [XMLSchema]
W3C (World Wide Web Consortium). *XML Schema*. World Wide Web Consortium, 2005 (see http://www.w3c.org/2001/XMLSchema)

W3C XML 1.0 [XML]
W3C (World Wide Web Consortium). Bray, Tim, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler & François Yergeau. *Extensible Markup Language (XML) 1.0* (Fourth Edition) World Wide Web Consortium, 2006. (see http://www.w3c.org/TR/2006/REC-xml-20060816/)

4 Terms and Definitions

For the purposes of this document, the following terms and definitions apply. Primary source for the terms in this section are taken or adapted from the references listed in Section 3 and the Bibliography in Section 7. This list of terms identifies conflicts between the meanings of terms commonly used in the speech-processing/SIV industries and that of similar terms used in the biometrics industry.

4.1 Active SIV Interaction / Active SIV Dialog

The process of obtaining spoken enrollment, verification, or identification utterances from a user who is aware of and actively involved in the SIV dialog. By extension, this includes Active Enrollment, Active Authentication, Active Identification, and Active Verification

4.2 Adaptation

The process of updating or refreshing a reference voice model. There are two primary kinds of model adaptation

1. Supervised adaptation is usually invoked by the application based on application or organization-specific criteria.
2. Unsupervised adaptation is typically performed automatically by the engine on a pre-determined basis.

4.3 Algorithm

A sequence of instructions that tell a biometric system how to solve a particular problem. An algorithm will have a finite number of steps and is typically used by the SIV system software to compute whether a voice model created from the speech of an end user and reference voice model are a match.

4.4 Analog Signal

An audio signal in which small changes in the input result in small changes in the output for both time and amplitude. It differs from a digital signal in that small fluctuations in the signal are meaningful. It is created in response to variations in air pressure resulting from speech and is produced using a transducer called a microphone. (Also see digital signal.)

4.5 ASR

An acronym for automatic speech recognition. (See speech recognition)\

4.6 Authentication

The process of confirming one or more identities. Synonymous with verification when a claim of identity has been made or identification when a group membership has been attempted and/or individual.

4.7 Bandwidth

This term has a number of uses. In this standard it refers to the frequency range of a transmission channel, such as wireline public service telephone network..

4.8 Base Standard

Fundamental and generalized procedures. They provide an infrastructure that may be used by a variety of applications, each of which may make its own selection from the options offered by them.

4.9 Biometric

A measurable, physical characteristic or personal behavioral trait used to recognize the identity, or verify the claimed identity, of an enrollee.

4.10 Biometric Data

The information extracted from the biometric sample and used either to build a reference model or to compare against a previously-created reference model. .

4.11 Biometric Sample

Raw data representing a biometric characteristic of an end-user as captured by a biometric system (for example the spoken utterances in one turn of a dialog). Normally, the biometrics industry refers to this as a "sample." That use of the word "sample" is not recognized by the SIV or speech-processing industry and creates confusion with the standard use of "sample" (see below).

4.12 Biometric System

An automated system capable of:

1. receiving one or more utterances from an end user;
2. extracting biometric data from those utterances;
3. comparing the biometric data with that contained in one or more reference models;
4. deciding how well they match and whether more data are needed;
5. generating a score that expresses the results of matching; and
6. indicating whether or not an identification or verification of identity has been achieved or communicating a final score to an application that makes a decision. .

4.13 Capture

The acquisition of a spoken sample.

4.14 Challenge-Response

A synonym for text prompted.

4.15 Claimant

A person submitting a biometric sample for verification or identification while claiming a legitimate or false identity.

4.16 Close Talking

When the microphone or telephone is positioned very close to the speaker's mouth, usually less than two inches. (Also see far field).

4.17 Co-located

SIV and ASR are combined in one engine. Typically they share some processing phases like endpointing, feature extraction etc. and have a set of shared return results like nonmatch/noinput.

4.18 Comparison

The process of comparing processed utterances with a previously stored reference model or models.

4.19 Database

Any storage of voice models and related end user information. Even if only one voice model or record is stored, the database will simply be "a database of one". Generally speaking, however, a database will contain a number of biometric records.

4.20 Dialog/ Dialogue

A dialog is the interaction between a user and computer during the length of a session

4.21 Digital Signal

An audio signal containing a discrete set of values that has been taken from an analog signal by sampling and by calculating an analog-to-digital conversion of the analog values in the original signal. Unlike an analog signal, a digital is a sequence of quantities. Each value in the sequence is called a sample. Also see analog signal, sample

4.22 Dynamic Challenge / Dynamic Text Prompting

A form of text prompting/challenge-response that dynamically generates words, digit sequences, or phrases and challenges the end user to repeat them or dynamically generates questions that the end user must answer. Typically, the end-user's responses have not been presented to the system in any previous session.

NOTE: This is not a formal SIV term but it defines a commonly-used approach to text prompting designed to enhance resistance to tape attacks.

4.23 Element and Entity

These are basic constituents of a well-formed XML document.

- An Element is a section of an XML document that generally begins with a start tag and concludes with an end tag. It is roughly equivalent to the concept of "field" in a binary CBEFF-compliant file format record.
- An Entity is a character string within an XML document that serves as a storage unit

Also see tag.

4.24 End User

A person who interacts with a biometric system to enroll or have his/her identity checked. This is distinguished from “User’. (See User)

4.25 Enrollment

The process of collecting voice samples from a person and the subsequent generation and storage of voice reference models associated with that person.

4.26 Far Field

The opposite of close talking. When the microphone or telephone is more than two inches away from the speaker’s mouth. This describes spoken input to speaker phones, hands-free phones, and array microphones.

4.27 Frequency / Cycles per Second

The measurement of the number of times that an individual sound wave repeats its cycle per second. It is often expressed as “cycles per second” or as “hertz.” Voices and other sounds that are not pure tones span many frequencies and are generally described in terms of their bandwidth.

4.28 Hertz

The hertz (Hz) is the international standard unit for expressing frequency. Its name comes from physicist Heinrich Hertz. Its base unit is s^{-1} (also called inverse seconds, or 1/s). In English, *hertz* is used as both singular and plural. Synonym for Frequency.

4.29 Identification

The one-to-many process of comparing a submitted biometric sample against all of the biometric reference templates on file to determine whether it matches any of the templates and, if so, the identity of the enrollee whose template was matched. The biometric system using the one-to-many approach is seeking to find an identity amongst a database rather than verify a claimed identity. Contrast with ‘Verification’.

4.30 Interactive Voice Response (IVR

An automated speech capture and output device used for SIV dialogs. It is usually written as IVR.

4.31 Liveness Detection

Technology or methods (e.g., challenge-response) designed to detect whether the end-user is a live human or a tape recorder being used to spoof the system

4.32 Matching

The process of comparing a voice sample against a previously stored reference voice model and scoring the level of similarity. An accept or reject decision is then based upon whether this score exceeds the given threshold.

4.33 Markup

The representation of elements, entities, and attributes in a well-formed XML document. It employs a formal syntax that is applied to the description of the storage layout and logical structure in the document. (also see element, entity and tag)

4.34 Microphone

An ambiguous term that can refer to

1. a standalone input device, such as a handheld microphone (called *non-telephony microphone* in this standard) or

2. the transducer component within those devices and telephones that converts changes in air pressure created by speech into an electrical signal.

4.35 Model

A processed representation of an end user's voice created by an SIV engine. It is derived from data extracted from one or more voice samples provided by that end user. Also called a Voice Model. It is rarely called a Template or Voice Template.

4.36 Passive SIV interaction / Passive SIV dialog

The process of obtaining spoken enrollment, verification, or identification from an end-user who may or may not be aware of the operation of the SIV system and who is not consciously interacting with SIV technology. By extension, this includes Passive Enrollment, Passive Authentication, Passive Identification, and Passive Verification.

4.37 Prompt

A request for action, for example audio response. An instruction (generally verbal) to the end-user that is designed to elicit utterances (e.g., "Please say your password"). It may emanate from the SIV system, the application, or a human with whom the end user is speaking.

4.38 Public service telephone network (PSTN)

The public telephone network.

4.39 Reference Model

The voice model that is stored and used by an SIV system for matching.

4.40 Replay Attack

Synonyms for Tape Attack**.**

4.41 Response

A reaction to a prompt; action may be an audio response.

4.42 Sample, Sampling Rate

A slice of the speech signal that is extracted from the signal during pre-processing. The number of samples taken per second is called the *Sampling Rate*, For an SIV may take 8,000 samples per second for speech transmitted via the PSTN.

4.43 SIV

An acronym for speaker identification and verification.

4.44 Speech Recognition/ASR

Non-biometric technology that decodes the words and phrases that the end-user has spoken. Also called Automated Speech Recognition.

4.45 Spoofing

Imitating the biometric of an authorized end-user (e.g., mimic, tape recorder).

4.46 Tag

The basic markup structure of a well-formed XML document.

4.47 Tape Attack

An attempt to spoof an SIV system by playing a recording of the end-user. Synonym for Replay Attack.

4.48 Template

A term used in the biometrics industry to refer to a model or reference model. This term is rarely used in the SIV industry. The terms Model, Voice Model, and Voiceprint are used instead.

4.49 Text dependent

SIV technology (usually verification technology) that requires the voice input of one or more specific passwords or pass phrases (having been enrolled).

4.50 Text independent

SIV technology that can process any kind of spoken input whether it is freeform or structured.

4.51 Text prompted

A kind of SIV dialog that generally involves a series of requests to the end-user to repeat randomly-selected sequences (e.g., “say 12345” followed by “say 43623”) or to answer randomly-selected questions (“what is today’s date”). One variant of this is dynamic text prompting. A synonym is challenge-response.

4.52 Turn /Interaction turn

A dialog with the user that consists of a single request and a single response. Synonymous with Interaction Turn.

4.53 User

The client of a biometric vendor. The user must be differentiated from the end user and is responsible for managing and implementing the biometric application rather than actually interacting with the biometric system. This is distinguished from End user (see End user)

4.54 Uniform Resource Indicator (URI).

A string of characters used to identify or name a resource (e.g., the VoiceXML Forum’s home page, the location where turn data are stored). A URI enables interaction with representations of the resource over a network, typically the Internet, using specific protocols. A URI is generally defined using specific syntax and associated protocols. For example, the VoiceXML Forum home page.

4.55 Uniform Resource Locator (URL)

A kind of URI that not only identifies a resource (e.g., the VoiceXML Forum's home page) but also supplies a means of accessing or locating the resource on the network. In this document, that generally means a World Wide Web location. For example, http://www.voicexml.org. is the URL of the VoiceXML Forum. It can be reached

4.56 Utterance

A spoken input speech sample. It may be real time streaming audio, a prerecorded file, or the result of buffering. In interactive systems, a single utterance typically corresponds to a single interaction turn. It is used instead of the terms "biometric sample" and "sample" to reflect common parlance within the SIV industry and to eliminate the ambiguity those other terms would produce.

4.57 Verification

The process of comparing a voice model created from a submitted utterance against the biometric reference model of a single enrollee whose identity is being claimed, to determine whether it matches the enrollee's reference model. Contrast with 'Identification'.

4.58 Voiceprint

A synonym for Model / Voice Model.

4.59 Voice Recognition

An ambiguous term that is both a synonym for "speech recognition" and a synonym for SIV. It is not used in this document and generally not used within the speech-processing industry to refer to SIV.

4.60 Voice Over IP / VoIP

Digitized streaming speech carried over data channels as IP packets.

4.61 Volume

A calculation of the "loudness" of the input signal (including speech) that the input source supports and is using. When it is known, it is expressed in terms of the International Telecommunications Union's P.56 algorithm. Volume level is a factor in the quality of the input utterances.

4.62 XML

An acronym for "Extensible Markup Language" which is a W3C-recommended general-purpose markup language that supports a wide variety of applications.

5 SIV Sessions

This section defines the fundamental elements of SIV interactions and the representation of end-user utterances/raw speech data captured during those interactions.

The fundamental organizing principle is the dialog. Compatible dialog structuring and acoustic signal description are required for interoperability between different SIV engines for the purposes of matching an individual against a previously collected and stored voice record. Interoperability is based on defining the interaction and acoustic rules that are common to many SIV engines for acceptable matching accuracy while allowing for extended data to be attached for use with equipment that is compatible with it. The rules and requirements of this standard are represented in XML ([XML]).

5.1 Dialogs

Establishment of a common representation of raw speech data depends upon agreement regarding the way in which SIV sessions are constructed. A significant number of SIV technology providers and integrators use the concept of "dialog" as that organizing principle. This principle is particularly applicable to the work being done by technology providers and integrators utilizing the VoiceXML standard language ([VXML]).

An SIV dialog is a verbal interaction for the purpose of biometric enrollment, verification and/or identification that is conducted by an end-user with an automated system or another human. The dialog may be an active or a passive SIV interaction. A dialog contains one or more "turns." Generally, a turn consists of a prompt to the end user requesting a response and the end-user's response. Figure 1 illustrates a simple verification dialog between an interactive voice response (IVR) system and an end user.

IVR: Welcome to the ABC Bank home-banking security system.
Please say your account number.
End user: 357128999
IVR: Thank you. Please say your password
End user: lollapalooza
IVR: Thank you.

Figure 1 Dialog 1: Basic verification dialog

The dialog in Figure 1 contains two turns.

Turn 1: IVR: Please say your account number.
End user: 357128999

Turn 2: IVR: Thank you. Please say your password
End user: lollapalooza

Figure 2 Turns in dialog 1

The first turn elicits a claim of identity from the end user and the second turn collects the voice data that will be used to verify the end-user's claim of identity. The dialog in Figure 1 would not need to change for end users interacting with humans (e.g., a call center agent). Multi-modal variants of dialog 1 include asking or allowing the end user to input the claim of identity (account number) manually (e.g., using the touchtone

keypad of the telephone). Prompts are normally presented as audio by playing one or more sound files or by generating a text-to-speech (TTS) output for an internal text string. Multi-modal applications may present the prompts as text displays (e.g., on PDAs).

From the end user's perspective the simplest active SIV dialog would contain only one turn. In dialog 1 this can be accomplished in two ways. Some applications use caller ID and/or other methods to implicitly establish the claim of identity. The result is a one-turn dialog (Turn 1 only). The dialog may also be reduced to a single turn (Turn 2 only) when automated speech recognition (ASR) is used. In that instance the IVR asks the end user to say the account number. ASR decodes the digits and uses them to retrieve the voice model. Then it sends the same input to the SIV engine for biometric verification.

As Figure 3 reveals, the same dialog and turn structure is also used for enrollment.

IVR: Welcome to the ABC Bank voice enrollment system.
Please say your account number.

End user: 357128999

IVR: Thank you. You will now be asked to repeat your password four times. After the tone, please say your password

[tone]

End user: lollapalooza

IVR: After the tone, please say your password again.

[tone]

End user: lollapalooza

IVR: After the tone, please say your password again.

[tone]

End user: lollapalooza

IVR: After the tone, please say your password again.

[tone]

End user: lollapalooza

IVR: Thank you. You are now enrolled in the ABC Bank voice security system.

Figure 3 Dialog 2: enrollment

.
This dialog contains five turns and the prompts for four of those turns include a tone.

5.2 Kinds of SIV

Each dialog turn has an SIV "type" associated with it. There are three major types of SIV:

- text dependent
- text prompted
- text independent.

With few exceptions, all three utilize the dialog structure described in the preceding section. The primary differences among these three SIV types involve the amount of speech needed to perform an SIV operation, the variability of the spoken material, and whether usable speech must be enrolled (dynamic vs. static material).

The dialogs in figures 1 through 3 are examples of “text-dependent” SIV dialogs. In a text-dependent dialog the end user is asked to say an enrolled password or pass phrase. Although text-dependent technology may ask for the end user’s name in most cases the item that is enrolled and used for verification is a shared secret. This means that authentication using text-dependent SIV technology is inherently two-factor authentication. The shared secret may be user-defined (e.g., BigBoy), system-defined, an existing identifier (e.g., an account number, social security number), or even a global password (e.g., “verification by Chemical Bank.”). The utterances are typically very short (usually around 2 seconds).

As Figure 3 reveals, the enrollment dialog consists of several to repetitions of the same password. Passive enrollment may span several sessions during which the end user is asked to supply the shared secret either by an IVR or by a human. The verification dialog generally consists of a single prompt to say the shared secret (e.g., “Tell me your password.”) followed by a single response. The dialog may continue for a pre-determined number of iterations if the interim score for the end-user’s input is low.

A “text-prompted” SIV interaction (also called “challenge-response”) involves one or more prompts to either repeat randomly-selected items/sequences or to answer randomly-selected questions. The enrollment dialog prompts for a series of different responses. Sometimes they are shared secrets (e.g., “Where were you born?”) and sometimes words, numeric sequences, natural numbers, phrases, or sentences (e.g., “Say Chicago, Illinois” “Say 54 35” “Say 1 2 3 4 5”). As with text-dependent SIV, text-prompted input is generally short. Figure 4 displays a text-prompted enrollment of patterns that are called “combination lock” sequences.

IVR: Welcome to the ABC Bank voice enrollment system.
Please say your account number.

End user: 357128999

IVR: Thank you.. After the tone, please say “thirty-four sixty-seven”

[tone]

End user: thirty-four sixty-seven

IVR: After the tone, please say “ninety-two eighty-three”

[tone]

End user: ninety-two eighty-three

IVR: After the tone, please say “twenty-one eighty-nine”

[tone]

End user: twenty-one eighty-nine

IVR: Thank you. You are now enrolled in the ABC Bank voice security system.

Figure 4 dialog 3: text-prompted enrollment

For authentication, the system randomly selects from among the enrolled items and the prompts the end user to repeat the item Figure 5 contains a typical text-prompted dialog for the Dialog 3 enrollment.

IVR: Welcome to the ABC Bank voice security system.
Please say your account number.
End user: 357128999
IVR: After the tone, please say "ninety-two eighty-three"
[tone]
End user: ninety-two eighty-three
IVR: Thank you.

Figure 5 Dialog 4: text-prompted verification using combination lock

The text-prompted dialog in Figure 6 would be used in a system that enrolled a series of questions.

IVR: Welcome to the ABC Bank voice security system.
Please say your account number.
End user: 357128999
IVR: What is your favorite color?
[tone]
End user: blue
IVR: Thank you.

Figure 6 Dialog 5: text-prompted verification using questions

The randomness in text-prompted dialogs makes text-prompting a strong anti-spoofing/liveness tool.

Text prompting may be used as the primary SIV type or as a secondary type when, for example, a tape attack is suspected or an interim matching score is questionable. Some SIV engines and applications employ a secondary text-prompted dialog that is often called "knowledge verification" following a text-dependent dialog that produced a poor matching score. This is why the format specifies that the type of SIV be indicated for each individual turn of an SIV dialog. This is also why it is important to know the content of prompts.

Text-dependent and text-prompted SIV should verify that the end user said what the system expected to hear. Systems that fail to validate the end user's input risk increased false-non-match/false-rejection errors. For example, a text-dependent system needs to verify that the end user said their password and not "I'm sorry. I didn't hear what you said." Many products and applications address this issue by using co-located speech recognition (ASR) or pattern matching technology. Some SIV require ASR and other co-located technologies because they need to know the language and language dialect that are being processed.

"Text-independent" SIV accepts freely-spoken input. It is the preferred approach for identification – especially passive identification. It focuses solely on biometric analysis, leaving content analysis to other technology, such as speech recognition (ASR) or pattern matching. Text-independent dialogs may be identical to that of other types of SIV but most of these systems require much longer blocks of speech. The purpose is to gather enough utterance data to create a voice model or to perform effective matching for verification or identification. This may range up to one minute of speech for enrollment and thirty seconds for verification, although those requirements are falling. For passive enrollment, verification, or

identification the technology simply listens in the background.

5.3 Capture and Use of Non-Enrolled Utterances

There are two ways SIV dialogs capture and use utterances that were not enrolled and, quite often, the end user has never said to the system in any prior session.

- Text-independent systems examine only the acoustic parameters of the end-user's voice. Consequently, they neither know nor care about what the person said.
- Dynamic challenge response (dynamic text prompting) asks the end user to say things that were not enrolled

Figure 7 provides an example of how dynamic text-prompting operates. Figure 7a contains an enrollment dialog for a dynamic text-prompting system. Figure 7b shows s a verification session. In 7b, the end user is prompted to say a sequence of digits that she/he did not say during enrollment and may never be asked to say again. Dynamic challenge enhances robustness against tape attacks.

IVR: Welcome to the ABC Bank voice security system. Please say your account number.

End user: 357128999

IVR: After the tone, please say "0 1 2 3 4 5 6 7 8 9"

[tone]

End user: 0 1 2 3 4 5 6 7 8 9

IVR: After the tone, please say "0 1 2 3 4 5 6 7 8 9"

[tone]

End user: 0 1 2 3 4 5 6 7 8 9

IVR: After the tone, please say "0 1 2 3 4 5 6 7 8 9"

[tone]

End user: 0 1 2 3 4 5 6 7 8 9

IVR: Thank you. You are now enrolled in the ABC Bank voice security system.

Figure 7a Dialog 6: enrollment for dynamic challenge Prompting

IVR: Welcome to the ABC Bank voice security system. Please say your account number.

End user: 357128999

IVR: After the tone, please say "3 6 9 7 5"

[tone]

End user: 3 6 9 7 5

IVR: Thank you.

Figure 7b Dialog 7: verification using dynamic challenge

Some dynamic challenge systems ask for time-linked responses, such as "What is today's date?" or "What was the amount of your last deposit to this account?" Dynamic challenge employs co-located or external ASR to validate the content of the utterance.

In this format both "text-independent" and "dynamic challenge" are listed alongside of "text dependent" and "text prompted" as separate types of SIV dialogs.

If this standard accepted standardized feature data rather than raw data text-independent and dynamic-challenge dialogs would provide the same kind of data captured by any other SIV dialog. As such, they be comparable to feature formats for any other biometric technology. In this raw-data format, however, they do not fit well into the existing CBEFF structure because CBEFF-compliant records are expected to contain models of what the end user is expected to produce. This is an issue that goes beyond the scope of this standard.

5.4 Input Devices

SIV technologies are designed to use input devices and channels that were created for purposes other than speaker authentication. Exceptions are extremely rare. The input devices include telephones and non-telephony microphones. Both contain transducers called “microphones” that convert the air pressure/sound wave patterns of speech into electrical signals. Those transducers vary in the ways they process audio signals. Furthermore, the spectrum of devices that can function as telephones (that is, communicate with PSTNs) is expanding rapidly as a result of the growth of VoIP. It is now commonplace to use a microphone embedded in a laptop, a headset microphone, or a handheld microphone as input to a VoIP telephone channel. The ability to process input from as many of these devices as possible is critical to the commercial success of SIV. That information is included in the format along with the audio format, compression, and related formatting information.

Microphone transducers of most devices perform best when used for close-talking input. Hands-free telephones, speaker phones, and array microphones are notable exceptions. Array microphones contain three or more microphone transducers that are configured to operate in synchrony to capture an audio signal. Their value lies in the ability to separate the acoustics of an auditory “sweet spot” from surrounding sounds. An array microphone might, for example, be used to capture speech at an entryway without requiring the speaker to use a close-talking device.

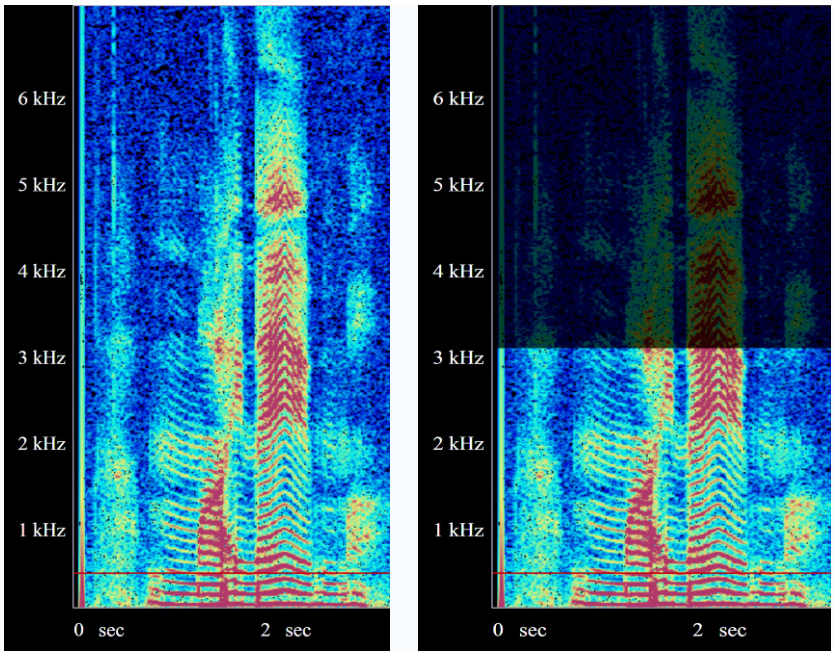
5.5 Channels

SIV engines and applications do not interact directly with input devices; they communicate with the channels that carry the signal to and from them. Like input devices, however, the channel carrying that signal is rarely a dedicated SIV channel, even though it might be a dedicated voice channel.

SIV systems can rarely access information about the input device but they can determine a number of important characteristics about the input channel. This includes the kind of channel it is (analog vs. digital; wireless vs. wireline), noise characteristics, the country (for local telephone calls), and channel capacity (bandwidth). Some of these attributes inform the SIV system about the kind of non-speech noise that is present or is likely to be present. For example, some SIV systems build acoustic models containing noise characteristics for different kinds of channels (primarily wireless vs. wireline) and use that model to separate the speech signal from the likely surrounding channel noise. Others estimate the signal-to-noise ratio (SNR) and may report that to the application as a quality control metric. Other systems require separate enrollments for each of the channels an end-user expects to use for authentication.

5.6 Bandwidth

The bandwidth of a channel is defined by the range of frequencies that it can carry. This definition is expressed in terms of the highest frequency the channel can carry (the “cutoff frequency”) and written as hertz (Hz), cycles per second (cps) or bits per second (bps). Wireline PSTNs have a standard bandwidth that ranges from 300 Hz/cps to slightly over 3100 Hz/cps (3.1 kHz). This bandwidth is often called the “telephone speech” bandwidth. Most non-telephony microphones and channels have a broader bandwidth than telephone speech.



a. **b.**

Figure 8 spectrogram of female saying "my name"

Figure 8 reveals the challenge that telephone speech represents for SIV (and ASR). Figure 8a shows that the frequency spectrum of the human voice extends at least to 6000 Hz (6 kHz). Figure 8b shows the bandwidth that is available to SIV engines working with speech captured and transmitted over a PSTN. The expanding bandwidth of new telephony devices and the advent of broad band channels are enhancing the quality of the input to SIV systems. Consequently, it is becoming increasingly important for an SIV engine to know the bandwidth of the channel with which it is communicating.

The concept of bandwidth extends to the input device because many input devices are often designed to be used with a specific channel. This is one reason why some SIV vendors express bandwidth for telephone channels in terms of categories of input devices (“wireline” or “wireless”). We use a direct numeric representation of bandwidth based on the cutoff frequency in this proposal rather than specifying a category of input devices. The reason is that channel bandwidth is growing rapidly and new input devices are being created to support broadband. This means that, “wireless telephone” could refer to a 4800 bps channel or a 64000 bps channel. Direct specification of bandwidth eliminates that ambiguity.

5.7 Sampling

Speech is a streaming medium. The size of the full speech signal is extremely large. Since the patterns present in the speech signal change slowly it is possible to reduce the processing requirements by using a subset of the whole signal. This is done by taking slices (called “samples”) from the stream. The sampling rate refers to the number of samples that are extracted from the audio stream per second.

The sampling rate affects the quality of the speech that is presented to the SIV engine. The more samples it receives the higher the quality of the speech. At the same time, there is a need to keep the size of the input and the processing required to a minimum. This tension has resulted in using a sampling rate that ensures that the cutoff frequency of the bandwidth is captured. That is accomplished by setting the sampling rate so that it will capture the start, middle, and end of the bandwidth cutoff frequency. This has been established to be a sampling rate of twice the frequency of the cutoff frequency ([NYQ]). The minimum sampling rate needed to capture telephone speech (bandwidth of 3100 kHz) is 6200 samples per second. Most SIV systems sample between 8 and 10 kHz.

must be extracted from the signal per second
is expressed in number of samples taken per second. For telephone-quality speech, that tends to be 8,000 samples per second. Input devices that can support input and transmission of higher-quality speech will use higher sampling rates.

5.7 Voice Model Adaptation

SIV applications and engines utilize adaptation to automatically update the information in a reference voice model. The added data may include information about the end user's voice, input devices, or channels. The reasons for adapting voice models are generally to unobtrusively gather additional information that will enhance the performance of the model or to counteract model aging. Systems that do not perform model adaptation may experience performance degradation over time.

There are two types of model adaptation: unsupervised and supervised. Unsupervised adaptation is done automatically following every successful authentication turn or session or at pre-determined regular intervals (e.g., every 3rd session). It is useful for gathering additional voice, input device, and channel characteristics for a designated period following enrollment.

Supervised adaptation is performed when the matching score for a turn or session is low but other factors strongly support the validity of the identity claim. For example, the end user may be calling on the same phone she/he always uses, they may demonstrate good knowledge of shared secrets, they may be engaging in a pattern of behavior that is typical for them, or they may be a combination of supportive information. The factors that are used and the weights assigned to them are part of the business and security rules and policies of the organization deploying the SIV application. Even when these rules apply there may be instances when adaptation is not performed. For example, adaptation may not be performed when the input is contaminated with noise.

Despite its value representing adaptation within this standard lies outside of the bounds of this standard because it operates on voice model data and this standard focuses on raw data transmission.

There are other concerns related to incorporating model adaptation into any standard SIV format. Unsupervised adaptation could be used by an impostor to modify the voice model towards the impostor's voice. Supervised adaptation provides greater confidence that the end user is who she/he claims to be but the rules and policies that permit or block adaptation may not be shared across organizations. This issue can be addressed through a formal agreement or other method that recognizes correspondence between the policies of the organization adapting a voice model and organizations using the adapted model.

Another problem with using an adapted voice model is that the model may be adapted towards or away from certain linguistic content, channel, or device. Those attributes are likely to differ between organizations and even between applications within a single organization. As a result, an adapting a reference voice model may result in poorer performance.

6 Voice Record Format

6.1 Introduction

The raw data voice record format (henceforth “VRR”) shall be used to achieve interoperability between and among SIV engines supporting both one-to-one verification and one-to-many identification. The dialog and turn data shall be represented in a common format containing both standardized fields/elements and data as specified by the dialog/record and turn portions of this format as well as non-standardized inclusions using the VRR extended data portion. Documents following this standard shall be written in XML 1.0 ([XML]).

6.2 XML

The markup language used for the VRR document and for documents using this standard is XML 1.0 ([XML], [XMLSchema]).XML is an acronym for Extensible Markup Language, a widely-used, standard markup-language created by the World Wide Web Consortium (W3C) as a simplified subset of Standard Generalized Markup Language (SGML; [ISO8879]). It is a general-purpose language that supports a wide variety of applications and is interoperable with SGML and HTML.

The primary goal of XML is to facilitate the sharing of data across different information systems, particularly systems connected via the Internet. The use of XML also enhances interoperability between this STANDARD and compliant documents with the W3C/ VoiceXML Forum’s VoiceXML 3.0 markup language [(VXML]) and the IETF’s version 2 Media Resources Control Language ([MRCP]). VoiceXML is the dominant markup language and MRCP is the dominant media control protocol in speech processing.

XML programs are textual constructs called “documents” that contain “elements” and “entities”. These components are described using the XML markup language. Markup describes the document’s storage layout and logical structure.

An element is a section of a well-formed XML document that begins with a start tag and concludes with an end tag (see Section 6.2.2). An element has a unique name and an associated data type. It may be further defined through the use of attributes. It is roughly equivalent to the concept of “field” in a binary CBEFF-compliant file format record.

An entity is a data storage unit. It may refer to content that appears within the document (e.g., a block of text that is repeated many times) or to content that exists outside of the document (e.g., an external file or binary data). Entities contain either parsed or unparsed data. Parsed data consists of character strings that can be interpreted (parsed) by XML. Those character strings may contain textual data, markup or a combination of textual data and markup. An unparsed entity is a resource whose contents cannot be analyzed by XML such as binary utterance data.

6.2.2 XML Document Structure

An XML document is “well formed” if it meets all of the well-formedness constraints in the XML standard (used correct syntax and semantics) and each of the parsed entities within it is also well formed.

A well-formed XML document has the following basic structure

- XML declaration
- Global declarations
- Root element.

6.2.3 XML Declaration

The XML declaration specifies which version of XML is being used. The following statement indicates that the document is using XML version 1.0.

<?xml version="1.0" ?>

The XML declaration may also indicate the encoding for the document:

<?xml version="1.0" encoding="UTF-8" ?>

This declaration says the document is using XML 1.0 and UTF-8 encoding.

The default encoding for XML documents is UTF-8 although XML 1.0 also explicitly supports UTF-16 and includes ISO10646 in its normative references. Use of other standard encodings, such as UTF-32, is not prohibited but it is not encouraged and must be explicitly declared whether it applies to the entire document or to a single entity within the document. This is in accordance with W3C's *Extensible Markup Language (XML) 1.0 (Fourth Edition)* which states on page 40 that

> Although an XML processor is required to read only entities in the UTF-8 and the UTF-16 encodings, it is recognized that other encodings are used around the world, and it may be desired for XML processors to read entities that use them. In the absence of external character encoding information (such as MIME headers) parsed entities which are stored in an encoding other than UTF-8 or UTF-16 MUST begin with a text declaration containing an encoding declaration…

6.2.4 Schema and Global Declarations

An XML document contains declarations that define global references, elements, and attributes in the document. One such declaration a VRR format must make is that it follows this standard. That is accomplished by the following declaration

<xsd:schema xmlns:xsd="Voice_VRR10.dtd">

This declaration references the external document "Voice_VRR10.dtd" which contains the XML specification of the requirements in this standard. The above declaration states that this is a voice raw data format (VRR) and that the version of the format is 1.0.

The sequence "xmlns:xsd" indicates that any elements or attributes with an "xsd:" prefix belong to the schema. Use of the "xsd" prefix is not necessary. Any prefix may be used as long as it is associated with the referenced schema but use of "xsd" with a named element, entity, or attribute that is not part of Voice_VRR10.dtd (or any other reference schema) is not permitted.

The "BIAS_Interface," is a schema created by Biometric Identity Assurance Service (BIAS) committee of ANSI INCITS M1. That schema includes an XML representation of the CBEFF header that is being developed by the ANSI INCITS M1 Biometric ([BIAS]). The current version of that header is presented in informative Annex B. Since The VRRs following this standard will be embedded in a CBEFF-compliant structure (Section 6.3) the BIAS_Interface Schema is included in the <schema> portion of Voice_VRR10.dtd.

The schema portion of a document also contains global elements and attributes. Voice_VRR10.dtd contains a global declaration for the attribute "byteOrder" which applies to binary content. It permits the use of Big-endian or Little-ending byte order. All multi-byte binary

data, such as utterance data, shall be represented as either big-endian or little-endian. This choice must be explicitly stated in the VRR. Big-endian organizes the bytes so that the most significant bytes are stored at lower addresses in memory than less significant bytes and transmits them in that sequence. Little-endian organizes the bytes so that the least significant bytes are stored at lower addresses in memory than more significant bytes and transmits them in that sequence.

6.2.4 Root Element

Each document contains a single "root element." The root element is the top-level element of the document following the prolog. It is the outermost element of the XML document and the "parent" of all other elements in the document. As shown below, the syntax for this root element follows the basic syntax used for other XML elements: it begins with a start tag concludes with an end tag.

Start tag: <SampleVRR>
:
content
:
End tag: </ SampleVRR>

6.3 Voice Record Organization

A voice record is called a "dialog." The XML sequence from Voice_VRR10.dtd states that a VRR format consists of a single CBEFF header and a single dialog.

```
<xsd:complexType name="VRRType">
      <xs:sequence>
            <xsd:element name="CBEFF_Header" type="CBEFF_BIR_Type"
               minOccurs="1" maxOccurs="1" />
            <xsd:element name="dialog"
               minOccurs="1" maxOccurs="1" />
      </ xsd:sequence>
```

The organization of a dialog is as follows:

- A single record header (dialog-header)
- Optional extended data for the dialog level (extended-data)
- One or more turns (turn)

which is expressed in XML AS?

```
<xsd:complexType  name="dialogType">
      <xsd:sequence>
            <xsd:element  name="dialog-header"
                  minOccurs="1" maxOccurs="1" />
            <xsd:element  name="extended_data"
                  minOccurs="0" maxOccurs="unbounded" />
```

```
            <xsd:element  name="turn"
                  minOccurs="1" maxOccurs="unbounded" />
      </ xsd:sequence>
```

Each extended-data portion has

- One or more extended-data sub headers
- Extended data for each sub header

Each turn consists of

- A single turn header
- Utterance data
- Optional extended data for that turn

The XML for these complex elements are in Annex 1.

The descriptions of these format portions is presented in the following sequence

- Dialog header
- Turn
- Extended data

6.4 Dialog Header

There shall be one dialog header for each VRR. The dialog header contains information about the characteristics of the dialog that will not change in the course of a single SIV dialog. Some of them are required and some are optional. They are show in Table 1.

Element Name	Status	Values
number-turns	Required	Integer
utterance-length-total	Required	Float in seconds
sampling-rate	Required	Integer, samples per second
audio-precision	Optional	Integer, in bits *(e.g., 8,12, 16)*
audio-format	Required	*GSM AMR* [GSM] *G.711* [G711] *G.722 (G.722.2 AMR-WB) (G.722.1)* [G722], *G.723.1* [G723] *G.726 16bit linear, DPCM, ADPCM* [PCM] [ADPCM] *G.728* [G728] *G.729 Annex A/Annex B /G.729AB*] [G729A] [G729B] [CELP AMR]
compression	Optional	Name of algorithm/codec default="None"
country	Optional	ISO 3166 Country code [CTRY1] [CTRY2]
voiceMF	Optional	*Female* *Male* default="unknown"
voiceInfo-dialog-level	Optional	Not defined
ED-dialog-length	Required	Size in bytes default="0"

Table 1 Dialog Header

The contents of Table 1 are represented in XML by the following element declaration for “dialog-header” described in Voice_VRR10.dtd:

6.4.1 number-turns

This is a required integer element specifying the number of interaction turns (see Sections 4.12, 4.52) included in the dialog. This is represented in XML by the following element declaration for “dialog” in Voice_VRR10.dtd.

```
<xsd:element  name=”number-turns”
              minOccurs="1" maxOccurs="1">
   <xsd:simpleType>
      <xsd:restriction base=”positiveInteger”>
      </ xsd:restriction>
   </xsd:simpleType>
</ xsd:element>
```

In a compliant VRR describing a dialog with five turns the information will be represented in the following way

<number-turns> 5 </number-turns>

6.4.2 utterance-length-total

This element is a companion to number-turns. It is a required float element specifying the total length in seconds of the end-user utterances from all the turns in the dialog. If the sum of the end-user utterances from the five-turn VRR dialog take 12 seconds of audio it will be represented in the following way

.

<utterance-length-total > 12.2 </utterance-length-total >

6.4.3 sampling-rate and audio-precision

Sampling-rate is a required integer element specifying the A numeric element indicating the number of samples taken from the speech signal per second.

Sampling-rate is required because speech is a streaming medium. SIV systems take slices (called “samples”) from the stream. Sampling rate is, therefore, a facet of the quality of the utterances. It is akin to the concept of pixels per inch in image processing.

<sampling-rate> 8000</ sampling-rate>

Audio-precision is an optional attribute of sampling-rate that refers to the width of samples in bits. Audio-precision varies from vendor to vendor and engine to engine but will not change within an engine. It is expressed in terms of the number of bits per byte. The most common audio precisions are 8- and 16-bit bytes and in most cases it is a facet of an audio format.

<sampling-rate> 8000 audio-precision=8</ sampling-rate>

This optional element has been included because some of the formats listed in audio-format do not define precision

6.4.4 audio-format and compression

Audio-format is a required text/string element. One of the variable characteristics of input devices is the audio format they use to encode speech before transmission. The most widely-used standard formats are listed in the table. The purpose of calling them out is to indicate the associated standards. Furthermore, the number and type of standard formats is likely to continue to change as telephones and transmission channels evolve. An audio-format statement would look like

<audio-format>"GSM"</ audio-format>

Compression specifies the compression algorithm used. Like JPEG in the image world, some formats include compression; others do not. This is why compression is an optional attribute of audio-format.

6.4.5 country

This is an optional string element specifying the country where the speech was collected. It references ISO 3166. It is useful for telephone calls with local originations because the noise and other characteristics of the telephone channel vary from country to country. Its utility is limited because there is also variability within countries and because the growth of VoIP and digital telephone networks that might originate from anywhere and could traverse national borders.

6.4.6 voiceMF

This is an optional string element specifying the SIV system's determination of the sex of the end-user. Most SIV vendors support it but there is disagreement about how well it functions.

6.4.7 voiceInfo-dialog-level

This is an optional element that was included because SIV vendors are developing features like "voiceMF" that further categorize the speaker and, thereby, enhance performance. This element is a placeholder that allows one or more of those features to be incorporated into the format as these features becomes commonplace. Since it is not yet defined, Voice-VRR10.dtd lists its XML data type is "ANY"

6.4.8 ED-dialog-length

This is a required integer element that allows the inclusion of extended data at the dialog level. It supports extended data, such as a language model or a voice model created from the attached end-user utterance data. It is equivalent to an object-tag in VoiceXML.

If its value is "0" then no extended data have been added to the dialog. A size greater than "0" specifies the length in bytes of the extended data tag for dialog-level.

Extended data at the dialog or turn levels is to be kept to a minimum and is not encouraged. In particular, inclusion of a voice model is discouraged because it can introduce security vulnerabilities.

6.5 Turn header and data

Element Name	Status	Values
number-channels	Required	Default= *mono* *stereo* Other (number)
volume	Optional	4 byte integer based on ITU-T P.56 algorithm [P56] -1=Unknown
SNR-estimate	Optional	Floating point numeric Default ="unknown"

		XML simple type “float”
bandwidth	Required	Integer in bits per second
Type-audioChannel	Required	*Analog* *Digital-non VoIP* *Digital VoIP* *Mixed* *Unknown*
speaking-distance	Optional	*close talking* *midfield* *farfield* *other* *unknown*
ASR-used	Required	Default=*yes* *no* *unknown*
language	Optional	ISO 639 language code Unknown [LANG1]
dialect	Optional	ISO 639-6 language dialect code [LANG2] Unknown
SIV-type	Required	*text dependent* *text-prompted*= challenge-response) *text independent* *dynamic challenge* -1=Unknown
dynamic-challenge	Optional	Not yet defined
prompt-content	Required	Pointer to location of prompt/prompt string or actual textual string. -1= Unknown
utterance-length	Required	Integer in seconds
utterance-content	Required	Pointer to location of text string or actual textual string for the utterance -1= Unknown
utterance	Required	Binary
EDTurn_length	Required	Size in bytes 0=default

Table 2 Turn

6.5.1 number-channels

This is a required string element that specifies the number of audio channels used in the transmission. It is included in the turn level because it is possible that the number of channels may change in the course of a dialog if, for example, they change telephones.

The number of channels has an impact on SIV processing. The default is “mono” because, today, most transmissions employ one “monaural” channel and are generally referred to as “mono.” There is, however, growing use of two or more channels using, for example, array microphones. Array microphones vary in the number of individual microphone receptors used.

6.5.2 volume

This is an optional integer element that defines the loudness that the input source is using. Volume level is a factor in the quality of the input utterances. It is optional because it may not always be known. When known, it is expressed in terms of a standard of the International Telecommunications Union’s P.56 algorithm. It appears at the turn level because it can be changed in the course of a dialog.

6.5.3 SNR-estimate

This is a required floating-point element that estimates the signal-to-noise ratio. The signal-to-noise ration is the relative loudness of the primary voice signal vs. that of the surrounding noise. It is best expressed as a ratio of numbers corresponding to decibel measurements. This standard reports that ratio as a single-precision, 32-bit, floating-point number. It is useful for assessment of quality but it is optional because many engines do not capture it.

6.5.4 bandwidth

This is a required integer element that specifies the channel capacity in terms of the highest frequency it can handle (called the "cutoff frequency"). It is not to be confused with sampling frequency, which expresses the number slices taken from the stream of speech for processing that is described in Section 6.4.3.

Bandwidth encapsulates a great deal of information related to both the input device (e.g., kind of telephone) and the channel. Vendors often present this information in terms of the input device (e.g., *wireless telephone, wireline telephone*) to make it easier for application developers to adjust prompts and call flows based on the overall channel characteristics. In some cases, those categories are really shorthand for the kind of channel and the channel capacity.

This standard does not include input-device categories in addition to bandwidth because channel bandwidth is expanding which, in turn, is producing new generations of input devices making any categorization of input devices difficult to maintain. For example, the term "wireless telephone" is already ambiguous based on the bandwidth that is supported and other device attributes.

6.5.5 type-audioChannel

This is a required string element that defines the kind of audio transmission channel used. Traditionally, "analog" was the dominant form but the rise of Voice Over IP (VoIP) has brought two significant changes that need to be communicated. One is that some channels are now entirely digital or known to be a mixture of digital and analog – whether they are wireline or wireless. The other change is that globalization of telephone networks has made it increasingly difficult to know what kind of channel is being used. For example, a telephone call originating and terminating in the US may be switched through a number of digital and analog networks that may not all be strictly within the US.

6.5.6 speaking-distance

This is an optional string element that indicates how far the end-user's mouth is from the input microphone. The actual distance cannot be specified so standard terms, such as "close talking," are used. It is optional because it is usually not available.

6.5.7 ASR-used

This is a required string element that indicates whether speech recognition (ASR) was used to analyze the content of the end-user's utterance. The default is "yes" because speech recognition is almost always used in VoiceXML applications of SIV and is frequently used in other SIV implementations. ASR may be a part of the SIV engine itself (called "co-located") or it may be provided by an outside ASR engine.

ASR is usually used to

- verify that the content of the utterances is what the system expects to receive
- reduce the number of turns in the dialog by applying ASR and SIV to the same utterance.

When ASR and SIV are both applied to the claim of identity the ASR decodes the content so that the SIV system can retrieve the correct voice model/template. This procedure is becoming extremely common.

ASR is not the only method of performing these operations. Other methods, such as general pattern matching, callerID, and other techniques may accomplish some of these purposes for which ASR is generally used.

6.5.8 language and dialect

Language and dialect are optional string elements that indicate the language and the language dialect used for the utterance. This element is included in at turn level rather than the dialog level because the end user may change language or dialect within a dialog and even within a single turn.

These fields are optional because most SIV technology is language blind. That is, it neither knows nor cares what language is being used. Other technology is language linked so it requires, at minimum, the language and may also need to know the dialect. The standards used are ISO codes for language and language region.

6.5.9 SIV-type

This is a required string element that identifies the type of SIV interaction that produced the utterance. SIV dialogs can involve more than one kind of interaction. For example, some engines may begin a dialog with a text-dependent challenge ("Say your password" or "What is your account number?") and proceed to a challenge-response/text-prompted interaction. That transition may occur automatically or as the result of other factors (e.g., insufficient amount of spoken data, questionable initial score, or suspicion of a tape attack).

6.5.10 dynamic-challenge

This is an optional element that is currently serving as a generic placeholder for yet undefined approaches for handling unenrolled, prompted input. As Section 5.3 discusses, dynamic challenge prompts the end user for utterances that have not been explicitly enrolled. It is a popular approach to authentication and a potent weapon against replay attacks. For that reason, the data type for this element is ANY.

This dynamism presents a challenge to the existing CBEFF structure because CBEFF-compliant formats are expected to contain the items requested as input (e.g., an enrolled fingerprint). This kind of dynamism would not be an issue in a format using feature or other intermediate representations.

6.5.11 prompt-content

This is a required element that contains the prompt that generated the utterance for the turn. It is important because it provides the context for understanding the user's response. Its data type is ANY because a prompt may consist of output from a text-to-speech synthesis system (string), playing of a pre-recorded audio (audio file), or a question posed by a human speaking with the end-user (streaming audio).

Issues of concern related to this element are:

1. The SIV engine may not know the content of the prompt and may not even care about it. This is particularly true for text-independent SIV engines – especially those operating in the background when the end-user is speaking with an ASR system or with another human.

2. There may be no "prompt" in situations where the SIV engine is monitoring the conversation of two or more human beings.

These are reasons the option "unknown" is included.

6.5.12 utterance-length

This is a required integer element specifying the length in seconds of the utterance data for the turn.

6.5.13 utterance-content

This is a required element that either contains the text of the utterance or specifies the location where a transliteration of the content resides. Since it supports more than one kind of data (e.g., text, pointer) its data type is ANY.

Having the content of the utterance can be important because some SIV engines can only handle certain kinds of content (e.g., digits only, or enrolled words). There are issues involved in capturing utterance content.

1. Some SIV engines have no way of knowing about utterance content because they have no speech recognition capabilities. As indicated in Section 5.3, text-independent SIV engines are generally not interested in the content of the response. They are only concerned with the relevant acoustic features of the response. They also generally have no way to capture the content.

2. Even with co-located ASR most SIV would not be able to provide a transcription of text-independent utterances because most ASR capture free-flowing speech from any speaker.

3. As with prompts, some utterances may be unique and will never be repeated (e.g., “What is today’s date?”). If all of the responses need to be stored for future use, then these responses will be of interest because of their acoustic attributes but they cannot be reused.

4. Many of the responses are shared secrets and must be protected. Sharing them – even for the purpose of interoperability – would violate privacy and render the individuals involved vulnerable to identity theft and other kinds of attacks.

These are all reasons why “unknown” has been included as a valid response. (Also, see the issues discussed in Section 6.5.11.).

6.5.14 utterance

This is a required element that is the actual end-user utterance for the turn. It takes an ANY data type because it may be presented as binary data or as a pointer to a storage location. In the CBEFF main header the data format is specified in the BioAPI_BIR_DATA_TYPE as “raw.”

One concern related to the use of a raw-data format is the transmission of raw data that contain shared secrets. They presents a higher security risk than transmission of intermediate or processed data and must be encrypted and otherwise secured.

The prompt data are not included in the turns because

1. The text of the prompt, if relevant, is likely to be included in prompt-content;
2. The prompt may have been dynamically generated by concatenation of multiple audio files and may not exist independently of the text;
3. The prompt may have been generated by a TTS system and, therefore, the text is the real source of the prompt; and/or
4. The SIV engine may not care what the prompt or the response is. It only wants a certain amount of data and it will tell the application when it has all the data it needs.

6.5.15 EDTurn-length

This is a required integer element is the turn-level correlate to ED-dialog-length. It allows the inclusion of

data, such as features, at the individual turn level. As with extended data at the dialog level, extended data is comparable to an object tag in VoiceXML. For more information on the extended data tag see Section 6.6.

Because this standard is designed to foster interoperability, raw utterance data must be included. Extended information and data are supported but their inclusion is strictly optional. They cannot replace the raw data and their inclusion is not encouraged. Security considerations for extended data should also be addressed or they may create security vulnerabilities.

6.6 Extended Data

The optional section of the VRR (voice raw-data) record is open to placing additional data both at the dialog and at the turn levels. The size of these sections shall be kept as small as possible, augmenting the data stored in the standard section. The extended data for the dialog level shall immediately follow ED-dialog-length; extended data for a turn shall immediately follow EDTurn-length element.

All records shall contain an element for extended data length at the end of the dialog header and the end of each turn. They are ED-dialog-length and EDTurn-length, respectively. These elements signal the presence or absence of extended data at that level. A value of zero will indicate that there is no extended data and that the file will end or continue with the next turn. A non-zero value will indicate the length of all extended data for that turn (or at the dialog level) and is followed by the extended-data tag which contains

- A single, required EDMain-Header
- One or more EDSub each of which consists of a header and extended data.

6.6.1 Common Extended Data Elements – EDMain-Header

EDMain-Header contains information about the extended data tag that is common to all kinds of extended data that might be appended to the dialog or a turn. It has two required elements.

Element Name	Values
EDNumber-subs	Integer
EDSubSize-Num#!	Integer (bytes)
⋮	
EDSubSize-Num#N	Integer (bytes)

Table 3 EDMain-Header

6.6.1.1 EDNumber-subs

This is a required integer element indicating the number of subtags (EDSub) that are appended to the current turn (or to the dialog level). More than one subtag might be needed to send two kinds of data (e.g., a set of features and an N-best list from ASR). Each of those kinds of data will have its own subtag within the extended data tag.

6.6.1.2 EDSubSize-Num##

This element specifies the size of each of the extended data subtags. There is one instance of the EDSubSize-Num element for each subtag. The purpose of this is to jump to a specific point within the current extended data tag or to skip the entire tag. The "##" represents the unique numeric identifier as specified in Section 6.6.2.1

NOTE: Each subtag has a unique numeric identifier constructed from the number of the turn (starting with 1) and the number of the subtag's its original place in the list of subtags for that turn (starting with 1). Subtags at the dialog level shall have identification numbers that begin with "0".

6.6.2 EDSub - extended data sub tags

The sub tag contains unique information related to the extended data along with the extended data. It contains four required fields.

Element Name	Values
EDSub-Tagnum	Integer
EDSub-Format	String
EDSub-VendorElement	ANY
EDSub-Segment	ANY

Table 4 Extended data sub tag

6.6.2.1 EDSub-TagNum

This integer element contains the unique numeric identifier of the subtag. It is constructed from the number of the turn (starting with 1) and the number of the subtag's original place in the list of subtags for that turn (starting with 1). Subtags at the dialog level begin with 0.

6.6.2.2 EDSub-Format

This string element identifies the format of the extended data in the subtag. The most common formats are binary, XML, URI, URL, and Vendor defined. The use of "vendor defined" is not encouraged because it can make it even more difficult for recipients to decode the extended data.

6.6.2.3 EDSub-VendorElement

This is a developer/vendor-defined element that characterizes the contents of the EDSub-Segment field. It may include text, pointers or other kind of data. For binary data, it may specify the byte order if that byte order is not big-endian or little-endian. Descriptions of feature data it may indicate the kind of features (e.g., LPC-derived cepstral, FFT coefficients, FFT-derived cepstral), window size and overlap, and other descriptors that facilitate parsing and utilization of the data in EDSub-Segment and Window size and/or overlap.

6.6.2.4 EDSubSegment

This is the extended data. The data must be in the format specified in EDSub-Format.

7 Bibliography

The following references provide additional information and insight into the provisions of this standard.

American National Standards Institute. *ATIS Telecom Glossary*. Washington, DC: American National Standards Institute, 2000.

Bagwell, Chris. *SoX* Sound Exchange. (see http://sox.sourceforge.net/), 2001.

Markowitz, Judith. *Using Speech Recognition* Upper Saddle River, NJ: Prentice Hall PTR, 1996.

H. Nyquist, *Certain topics in telegraph transmission theory*, Trans. AIEE, vol. 47, pp. 617-644, (Reprint as classic paper in: *Proc. IEEE, Vol. 90, No. 2, Feb 2002*). April, 1928.

Rabiner, Lawrence & Biing-Hwang Juang. *Fundamentals of Speech Recognition.* Saddle River, NJ: Prentice Hall PTR, 1993.

Swayze, Matt. BIAS - *INCITS Project 1823-D Draft Revision 2* (Document M1/06-0888), August, 2006.

VoiceXML Forum. *Speaker Identification and Verification (SIV) Glossary, 2006.*

Wikipedia, the Free Encyclopedia. Wikimedia Foundation, Inc. 2006. (see http://en.wikipedia.org/wiki/Main_Page)

Annex A (normative) – Voice_VRR10.dtd Schema

This is the schema to be referenced by VRR documents.

```
<?xml version="1.0" encoding="utf-8" ?>

<xsd:schema id="Voice_VRR10.dtd"
      xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xs="Bias_Interface"

<xsd:attribute name="byteOrder">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="big-endian"/>
            <xsd:enumeration value="little-endian"/>
        </ xsd:restriction>
      </ xsd:simpleType>
</ xsd:attribute>

<schema/>

<xsd:element name="VRRType">
<xsd:element name="extended-data" />
<xsd:element name="UtteranceType"
   type=ANY  ref="byteOrder" minOccurs="1" maxOccurs="unbounded"/>
```

```
      <xsd:complexType name="VRR">
            <xs:sequence>
              <xsd:element name="CBEFF_Header" type="CBEFF_BIR_Type"
                   minOccurs="1" maxOccurs="1"  />
              <xsd:element name="Dialog" type="dialogType"
                  minOccurs="1" maxOccurs="1"  />
            </ xsd:sequence>
<!--BIAS_Interface has been declared a schema of this document.
CBEFF_BIR_Type is shown in Annex B -->

<!-- dialog, is a child of VRR  -->
      <xsd:element name="dialog" type="dialogType"
            minOccurs="1" maxOccurs="1" />


      <xsd:complexType  name="dialogType">
            <xsd:sequence>
                  <xsd:element  name="dialog-header"
                        minOccurs="1" maxOccurs="1" />
                  <xsd:element  name="extended_data"
                        minOccurs="0" maxOccurs="unbounded" />
                  <xsd:element  name="turn"
                        minOccurs="1" maxOccurs="unbounded" />
            </ xsd:sequence>


<!-- dialog header, a child of dialog -->

       <xsd:complexType  name="dialog-header">
            <xsd:sequence>
                  <xsd:element  name="number-turns"
                        minOccurs="1" maxOccurs="1">
                     <xsd:simpleType>
                        <xsd:restriction base="positiveInteger">
                        </ xsd:restriction>
                    </xsd:simpleType>
                  </ xsd:element>
                  <xsd:element name="utterance-length-total"
                              minOccurs="1" maxOccurs="1" >
                        <xsd:simpleType>
                           <xsd:restriction base="xsd:"float">
                           </xsd:restriction>
                        </xsd:simpleType>
                  <xsd:element  name="sampling-rate" type="duration"
                          minOccurs="1" maxOccurs="1" />
                        <xsd:simpleType>
                           <xsd:restriction base="xsd:positiveInteger">
                           </xsd:restriction>
                        </xsd:simpleType>
                        <xsd:attribute name="audio-precision"
                           type="xsd:positiveInteger" use="optional" />
                  </xsd:element>
                  <xsd:element  name="audio-format"
                        type="xsd:string" minOccurs="1" maxOccurs="1" />
                     <attribute  name="compression"
                        Type="xsd:string" use="optional" />
```

```
                </xsd:element>
                <xsd:element  name=”country”
                      type="xs:string" minOccurs="0" maxOccurs="1" />
                <xsd:element  name=”voiceMF”  minOccurs="0" maxOccurs="1" />
                   <xsd:simpleType>
                     <xsd:restriction base=”xsd:string”>
                       <xsd:enumeration value=”female”/>
                       <xsd:enumeration value=”male”/>
                       <xsd:enumeration value=”unknown”/>
                     </ xsd:restriction>
                   </ xsd:simpleType>
                </ xsd:element>
                <xsd:element  name=”VoiceInfo-dialog-level”
                      Type=ANY minOccurs="0" maxOccurs="1" />
              <xsd:element  name=”ED-dialog-length”
                      type="xs:string" minOccurs="1" maxOccurs="1" />
           </xsd:sequence>
     </xsd:complexType>

<!-- turn, a child of dialog -->

      <xsd:complexType  name=”Turn”>
            <xsd:sequence>
                <xsd:element  name=”number-channels”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”volume”
                      type=”xsd:string” minOccurs=”0” maxOccurs=”1” />
                <xsd:element  name=”SNR-estimate”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”bandwidth”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”Type-audioChannel”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”Speaking-distance”
                      type=”xsd:string” minOccurs=”0” maxOccurs=”1” />
                <xsd:element  name=”ASR-used”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”language”
                      type=”xsd:string” minOccurs=”0” maxOccurs=”1” />
                <xsd:element  name=”dialect”
                       type=”xsd:string” minOccurs=”0” maxOccurs=”1” />
                <xsd:element  name=”SIV-type”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”dynamic-challenge”
                      type=ANY minOccurs=”0” maxOccurs=”1” />
                <xsd:element  name=”prompt-content”
                      type=ANY minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”utterance-length”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”utterance-content”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”utterance”
                      type=ANY minOccurs=”1” maxOccurs=”1” />
                <xsd:element  name=”EDTurn-length”
                      type=”xsd:string” minOccurs=”1” maxOccurs=”1” />
            </xsd:sequence>
```

```
      </xsd:complexType>

<!-- extended-data, used in both dialog and turn -->

      <xsd:complexType  name=”extended-data”>
            <xsd:sequence>
                  <xsd:element  name=”EDMain-Header”
                        minOccurs=”1” maxOccurs=”1” />
                  <xsd:element  name=”EDSub”
                        minOccurs=”1” maxOccurs=”unbounded” />
            </ xsd:sequence>
     </xsd:complexType>

      <xsd:complexType  name=”EDMain-Header” />
            <xsd:sequence>
                  <xsd:element  name=”EDNumber-subs”
                        type="xsd:string" minOccurs="1" maxOccurs="1" />
                  <xsd:element  name=”EDSubSize-Num” type=”xsd:string”
                        minOccurs=”1” maxOccurs=”unbounded”>
            </ xsd:sequence>
      </xsd:complexType>

      <xsd:complexType  name=”EDSub” />
            <xsd:sequence>
                  <xsd:element  name=”EDSub-Tagnum”
                        type="xs:string" minOccurs="1" maxOccurs="1" />
                  <xsd:element  name=”EDSub-Format”
                       type="xs:string" minOccurs="1"
                       maxOccurs="unbounded" />
                  <xsd:element  name=”EDSub-VendorElement”
                        type=ANY minOccurs="1" maxOccurs="1" />
                  <xsd:element  name=”EDSub-Segment”
                        type=ANY minOccurs="1" maxOccurs="1" />
             </xsd:sequence>
         </xsd:complexType>
```

Annex B (informative) – BIAS CBEFF Header

From Swayze, Matt. BIAS - *INCITS Project 1823-D Draft Revision 2* (Document M1/06-0888), August, 2006. This is the portion of the BIAS_Interface schema that defines the CBEFF header. Once it is approved it should be referenced in VRR documents.

```
<xs:complexType name="CBEFF_BIR_Type">
      <xs:sequence>
            <xs:element name="BDBSecurityAndEncyrptionOptions"
                  type="xs:string" minOccurs="1" maxOccurs="1" />
            <xs:element name="BIRIntegrityOptions"
                  type="xs:string" minOccurs="0" maxOccurs="1" />
            <xs:element name="HeaderVersion"
                  type="xs:string" minOccurs="0" maxOccurs="1" />
            <xs:element name="PatronHeaderVersion" type="xs:string"
                  minOccurs="1" maxOccurs="1" />
            <xs:element name="BiometricType"
                  type="xs:string" minOccurs="0" maxOccurs="1" />
            <xs:element name="BiometricSubType"
```

```
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BiometricDataType"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BDBPurpose" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="BDBQuality" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="BDBCreationDate"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BIRCreationDate"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BDBValidityPeriod"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BIRValidityPeriod"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BIRCreator" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="BDBIndex" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="BIRIndex" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="BDBChallengeResponse"
      type="xs:string" minOccurs="0" maxOccurs="1" />
<xs:element name="BIRPayload" type="xs:anyType"
      minOccurs="0" maxOccurs="1" />
<xs:element name="SubheaderCount"
      type="xs:positiveInteger" minOccurs="0" maxOccurs="1" />
<xs:element name="BDBFormatOwner" type="xs:string"
      minOccurs="1" maxOccurs="1" />
<xs:element name="BDBFormatType" type="xs:string"
      minOccurs="1" maxOccurs="1" />
<xs:element name="BDBProductIdentifier"
      minOccurs="0" maxOccurs="1">
   <xs:complexType>
      <xs:sequence>
            <xs:element name="BDBProductOwner" type="xs:string"
                  minOccurs="1" maxOccurs="1" />
            <xs:element name="BDBProductType" type="xs:string"
                  minOccurs="1" maxOccurs="1" />
      </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="BIRPatronFormatIdentifier"
      minOccurs="0" maxOccurs="1">
   <xs:complexType>
      <xs:sequence>
            <xs:element name="BIRPatronFormatOwner"
                  type="xs:string" minOccurs="1" maxOccurs="1" />
            <xs:element name="BIRPatronFormatType"
                  type="xs:string" minOccurs="1" maxOccurs="1" />
      </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="SBFormatOwner" type="xs:string"
      minOccurs="0" maxOccurs="1" />
<xs:element name="SBFormatType" type="xs:string"
      minOccurs="0" maxOccurs="1" />
```

```
            <xs:element name="Other" minOccurs="0" maxOccurs="unbounded">
               <xs:complexType>
                  <xs:sequence>
                        <xs:any namespace="##any" processContents="lax"
                              minOccurs="0" maxOccurs="unbounded" />
                        </xs:sequence>
               </xs:complexType>
            </xs:element>
            <xs:element name="BIR" type="xs:base64Binary"
                  minOccurs="1" maxOccurs="1" />
      </xs:sequence>
</xs:complexType
```