Network Working Group
Internet Draft
Document: draft-burger-speechsc-reqts-00.txt
Category: Informational
Expires August 2002

E. Burger
SnowShore Networks, Inc.
D. Oran
Cisco Systems, Inc.

June 13, 2002

Requirements for Distributed Control of ASR, SV and TTS Resources

Status of this Memo

1. Abstract

This document outlines the needs and requirements for a protocol to control distributed speech processing of audio streams. By speech processing, this document specifically means automatic speech recognition, speaker verification and text-to-speech. Other IETF protocols, such as SIP and RTSP, address rendezvous and control for generalized media streams. However, speech processing presents additional requirements that none of the extant IETF protocols address.

Discussion of this and related documents is on the MRCP list. To subscribe, send the message "subscribe mrcp" to majordomo@snowshore.com. The public archive is at http://flyingfox.snowshore.com/mrcp_archive/maillist.html.

NOTE: This mailing list will be superseded by an official working group mailing list, cats@ietf.org, once the WG is formally chartered.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].

FORMATTING NOTE: Notes, such at this one, provide additional, nonessential information that the reader may skip without missing anything essential. The primary purpose of these non-essential notes is to convey information about the rationale of this document, or to place this document in the proper historical or evolutionary context. Readers whose sole purpose is to construct a conformant implementation may skip such information. However, it may be of use to those who wish to understand why we made certain design choices.

OPEN ISSUES: This document highlights questions that are, as yet, undecided as "OPEN ISSUES".

3. Introduction

There are multiple IETF protocols for establishment and termination of media sessions (SIP[3]), low-level media control (MGCP[4] and MEGACO[5]), and media record and playback (RTSP[6]). This document focuses on requirements for one or more protocols to support the control of network elements that perform Automated Speech Recognition (ASR), speaker verification (SV), and rendering text into audio, a.k.a. Text-to-Speech (TTS). Many multimedia applications can benefit from having automatic speech recognition (ASR) and text-to-speech (TTS) processing available as a distributed, network resource. This requirements document limits its focus on the distributed control of ASR, SV and TTS servers.

To date, there are a number of proprietary ASR and TTS API's, as well as two IETF drafts that address this problem [7] [8]. However, there are serious deficiencies to the existing drafts. In particular, they mix the semantics of existing protocols yet are close enough to other protocols as to be confusing to the implementer.

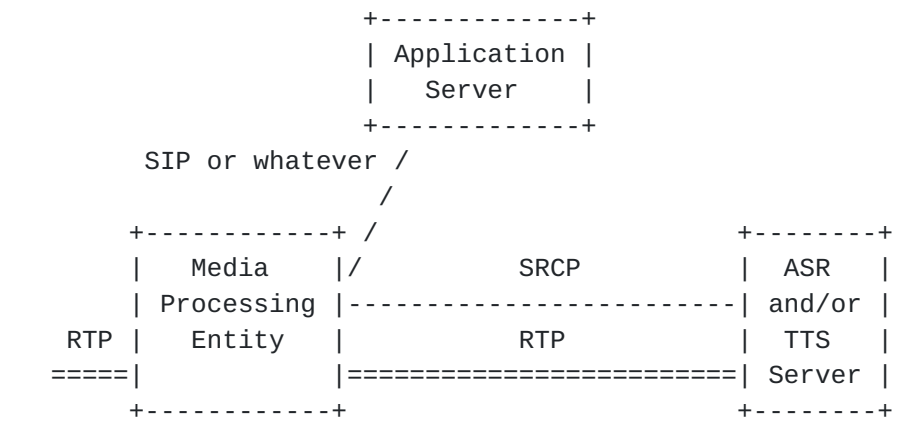
This document sets forth requirements for protocols to support distributed speech processing of audio streams.

For simplicity, and to remove confusion with existing protocol proposals, this document presents the requirements as being for a "new protocol" that addresses the distributed control of speech resources It refers to such a protocol as "SRCP", for Speech

Resource Control Protocol.

4. SRCP Framework

The following is the SRCP framework for speech processing.



The "Media Processing Entity" is a network element that processes media. The "Application Server" is a network element that instructs the Media Processing Entity on what transformations to make to the media stream. The "ASR and/or TTS Server" is a network element that either generates a RTP stream based on text input (TTS) or returns speech recognition results in response to an RTP stream as input (ASR). The Media Processing Entity controls the ASR or TTS Server using SRCP as a control protocol.

Physical embodiments of the entities can reside in one physical instance per entity, or some combination of entities. For example, a VoiceXML [9] Gateway may combine the ASR and TTS functions on the same platform as the Media Processing Entity. Note that VoiceXML Gateways themselves are outside the scope of this protocol.

Likewise, one can combine the Application Server and Media Processing Entity, as would be the case in an interactive voice response (IVR) platform.

One can also decompose the Media Processing Entity into an entity that controls media endpoints and entities that process media directly. Such would be the case with a decomposed gateway using MGCP or megaco. However, this decomposition is again orthogonal to the scope of SRCP.

5. General Requirements

5.1. Reuse Existing Protocols

To the extent feasible, the SRCP framework SHOULD use existing protocols.

5.2. Maintain Existing Protocol Integrity

In meeting requirement 5.1, the SRCP framework MUST NOT redefine the semantics of an existing protocol.

Said differently, we will not break existing protocols or cause backward compatibility problems.

5.3. Avoid Duplicating Existing Protocols

To the extent feasible, SRCP SHOULD NOT duplicate the functionality of existing protocols. For example, SIP with msuri [10] and RTSP already define how to request playback of audio.

The focus of SRCP is new functionality not addressed by existing protocols or extending existing protocols within the strictures of requirement 5.2.

5.4. Explicit invocation of services

The SRCP framework MUST be compliant with the IAB OPES[11] framework. The applicability of the SRCP protocol will therefore be specified as occurring between clients and servers at least one of which is operating directly on behalf of the user requesting the service.

5.5. Server Location and Load Balancing

To the extent feasible, the SRCP framework SHOULD exploit existing schemes for performing service location and load balancing, such as the Service Location Protocol[12] or DNS SRV records[13]. Where such facilities are not deemed adequate, the SRCP framework MAY define additional load balancing techniques.

6. TTS Requirements

The SRCP framework MUST allow a Media Processing Entity, using a control protocol, to request the TTS Server to playback text as voice in an RTP stream.

The TTS Server MUST support the reading of plain text. For reading plain text, the language and voicing is a local matter.

The TTS Server SHOULD support the reading of SSML [14] text.

OPEN ISSUE: Should the TTS Server infer the text is SSML by detecting a legal SSML document, or must the protocol tell the TTS Server the document type?

The TTS Server MUST accept text over the SRCP connection for reading over the RTP connection. The server MUST accept text either ?by value? (embedded in the protocol), or ?by reference? (by de-referencing a URI embedded in the protocol).

OPEN ISSUE: Should we allow (or require) the TTS Server to use long-lived control channels?

The TTS Server SHOULD support, and the SRCP framework MUST support the specification of, "VCR Controls", such as skip forward, skip backward, play faster, and play slower.

OPEN ISSUE: Should we allow for session parameters, like prosody and voicing, as is specified for MRCP over RTSP [7]?

OPEN ISSUE: Should we allow for speech markers, as is specified for MRCP over RTSP [7]?

7. ASR Requirements

The SRCP framework MUST allow a Media Processing Entity to request the ASR Server to perform automatic speech recognition on an RTP stream, returning the results over SRCP.

The ASR Server MUST support the XML specification for speech recognition [15].

The ASR Server MUST accept grammar specifications either ?by value? (embedded in the protocol), or ?by reference? (by de-referencing a URI embedded in the protocol).

OPEN ISSUE: Should we allow the ASR Server to support alternative grammar formats? If so, we need mechanisms to specify what format the grammar is in, capability discovery, and handling unsupported grammars.

OPEN ISSUE: Is there a need for all of the parameters specified for MRCP over RTSP [7]? Most of them are part of the W3C speech recognition grammar.

The ASR Server SHOULD support a method for capturing the input media stream for later analysis and tuning of the ASR engine.
The ASR Server SHOULD support sharing grammars across sessions. This supports applications with large grammars for which it is unrealistic to dynamically load. An example is a city-country grammar for a weather service.

8. Speaker Verification Requirements

The SRCP framework MUST allow a Media Processing Entity to request the SV Server to perform speaker verification on an RTP stream, returning the results over SRCP.

The SV Server MUST The server MUST accept grammar specifications either ?by value? (embedded in the protocol), or ?by reference? (by de-referencing a URI embedded in the protocol).

The SRCP framework MUST accommodate an identifier for each verification resource and permit control of that resource by ID, because voiceprint format and contents are vendor specific

The SRCP framework MUST work with SV servers which maintain state to handle multi-utterance verification.

The SV Server SHOULD support a method for capturing the input media stream for later analysis and tuning of the SV engine.

9. Dual-Mode Requirements

One very important requirement for an interactive speech-driven system is that user perception of the quality of the interaction depends strongly on the ability of the user to interrupt a prompt or rendered TTS with speech. Interrupting, or barging, the speech output requires more than energy detection from the user's direction. Many advanced systems halt the media towards the user by employing the ASR engine to decide if an utterance is likely to be real speech, as opposed to a cough, for example.

To achieve low latency between utterance detection and halting of playback, many implementations combine the speaking and ASR functions. The SRCP framework MUST support such dual-mode implementations.

10. Thoughts to Date (non-normative)

The protocol assumes RTP carriage of media. Assuming session-

oriented media transport, the protocol will use SDP to describe the session.

The working group will not be investigating distributed speech recognition (DSR), as exemplified by the ETSI Aurora project. The working group will not be recreating functionality available in other protocols, such as SIP or SDP.

TTS looks very much like playing back a file. Extending RTSP looks promising for when one requires VCR controls or markers in the text to be spoken. When one does not require VCR controls, SIP in a framework such as Network Announcements [16] works directly without modification.

ASR has an entirely different set of characteristics. For barge-in support, ASR requires real-time return of intermediate results. Barring the discovery of a good reuse model for an existing protocol, this will most likely become the focus of SRCP.

11. Security Considerations

Protocols relating to speech processing must take security into account. This is particularly important as popular uses for TTS include reading financial information. Likewise, popular uses for ASR include executing financial transactions and shopping.

We envision that rather than providing application-specific security mechanisms in SRCP itself, the resulting protocol will employ security machinery of either containing protocols or the transport on which it runs. For example, we will consider solutions such as using TLS for securing the control channel, and SRTP for securing the media channel.

12. References

1 Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

2 Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997

3 Handley, M., Schulzrinne, H., Schooler, E., and Rosenberg, J., "SIP: Session Initiation Protocol", RFC 2543, March 1999

4 Arango, M., Dugan, A., Elliott, I., Huitema, C., and Pickett, S., "Media Gateway Control Protocol (MGCP) Version 1.0", RFC 2705, October 1999

5 Cuervo, F., Greene, N., Rayhan, A., Huitema, C., Rosen, B., and Segers, J., "Megaco Protocol Version 1.0", RFC 3015, November 2000

6 Schulzrinne, H., Rao, A., and Lanphier, R., "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998

7 Shanmugham, S., Monaco, P., and B. Eberman, "MRCP: Media Resource Control Protocol", draft-shanmugham-mrcp-01.txt, November 2001, work in progress

8 Robinson, F., Marquette, B., and R. Hernandez, "Using Media Resource Control Protocol with SIP", draft-robinson-mrcp-sip-00.txt, September 2001, work in progress

9 World Wide Web Consortium, "Voice Extensible Markup Language (VoiceXML) Version 2.0", W3C Working Draft, <http://www.w3.org/TR/2001/WD-voicexml20-20011023/>, October 2001, work in progress

10 Van Dyke, J. and Burger, E., "SIP URI Conventions for Media Servers", draft-burger-sipping-msuri-01, July 2001, work in progress (expired)

11 Floyd, S., Daigle, L., ?IAB Architectural and Policy Considerations for Open Pluggable Edge Services,? RFC3238, January 2002.

12 Guttman, E., Perkins, C., Veizades, J., Day, M. , "Service Location Protocol, Version 2,? RFC 2608, June 1999.

13 Gulbrandson, A, Vixie, P., Esibov, L., ?A DNS RR for specifying the location of services (DNS SRV)?, RFC2782, February 2000.

14 World Wide Web Consortium, "Speech Synthesis Markup Language Specification for the Speech Interface Framework", W3C Working Draft, <http://www.w3.org/TR/speech-synthesis>, January 2001, work in progress

15 World Wide Web Consortium, "Speech Recognition Grammar Specification for the W3C Speech Interface Framework", W3C Working Draft, <http://www.w3.org/TR/speech-grammar/>, August

2001, work in progress

16 O'Connor, W., Burger, E., "Network Announcements with SIP", draft-ietf-sipping-netann-01.txt, November 2001, work in progress

13. Acknowledgments

Brian Eberman came up with the new name. It is catchy and describes what we are working on.

14. Author's Addresses

Eric W. Burger
SnowShore Networks, Inc.
Chelmsford, MA
USA
Email: eburger@snowshore.com

David R. Oran
Cisco Systems, Inc.
Acton, MA
USA
Email: oran@cisco.com

15. Change Log

From version draft-burger-mrcp-reqts-00 to version draft-burger-speechsc-reqts-00:

- draft name changed per area director advice
- added speaker verification to the areas addressed, including speaker verification requirements, per Dan Burnet?s presentation at the Minneapolis BoF (see minutes).

- based on mailing list discussion, added requirement to handle both ?by value? and ?by reference? data. This is both for TTS to be played out and grammar(s) to be applied to ASR.
- Based on discussion at the BoF in Minneapolis, added a requirement concerning the use of load balancing schemes, including those based on SRVLOC, SRV.
- Added a requirement for OPES compliance, per a discussion with Sally Floyd as IAB observer for the BoF.

Full Copyright Statement

Acknowledgement

The Internet Society currently provides funding for the RFC Editor function.