

## **Impact of standards on Automatic Speech Recognition**

*Paolo Baggia, Loquendo*

The aim of this article is to explain the impact of today's standards on speech recognition (ASR). The main areas covered are the results of the work done in the World Wide Web Consortium (W3C) Voice Browser Working Group [1], and also the Internet Engineering Task Force (IETF) [2]. The paper is organized in dialogue form to answer the most frequent questions on standards in ASR.

### ***Why standards for ASR grammars?***

The use of standards in ASR grammar can lead to great advantages, such as to increase portability, to speed up development and aid grammar reutilization. Standards can also reduce the workload of developers during both syntax learning and grammar coding.

### ***But what is a speech recognition grammar, in short?***

It is a way of describing, in a concise and efficient way, all the possible sentences that may be spoken at a certain point in a speech interaction dialog. It may be a list of speakable sentences or words (as in the most simple case of a very focused question: "Which is the arrival's station?"), or a more complex set of rules that describes, at different levels, sentences that might be spoken.

A grammar is organized into rules: the main one is called the "root" rule for the grammar, and it acts as an entry point inside the grammar itself. A rule can reference other rules, or list combinations of equivalent alternative wordings, repetitions or optional parts. The grammar developers design the rules to organize a grammar in an efficient and maintainable way.

### ***Which are the main standards for ASR grammars?***

There is only one standard for ASR grammars: it is "Speech Recognition Grammar Specification", called SRGS in short ([3]). It was given W3C Recommendation in March 2004. This means that many companies demonstrated it to be easy to implement, and gave support to its development. SRGS includes two different formats to encode a speech recognition grammar, they are:

- An XML format with an enforced syntax expressed both by a DTD and a Schema.
- An ABNF (Augmented Backus-Naur Form) format, which is a textual and concise encoding of a grammar

### ***What are the benefits of two formats?***

It is worth noting that the two formats of SRGS are equivalent, which means that a grammar expressed in XML can be transformed into ABNF format and possibly turned back into XML. These changes will not impact on the grammar itself, but only in the way the grammar is presented to the developer.

The two formats cope with different needs: ABNF format is very suitable for quick hand coding, while XML is easily handled in automatic environments and is more suitable for integrating into XML based VUI design languages, i.e. VoiceXML 2.0 ([11]).

These peculiarities of SRGS grammars allow the grammar developers to choose the format they are more accustomed to, with a potential reduction in the cost of maintenance of ASR grammar resources.

A second benefit is that it promotes the creation of standard grammar development tools, that allow different users to interact with the single grammar in the most effective way.

### ***Are there other powerful features in SRGS?***

Yes. The most powerful feature is the use of URI (a more correct term for the common URL, like http or ftp, used in a web-browser) for indicating a grammar rule. This simple fact allows the

grammar to be located on the web and to be downloaded and compiled when needed. It also allows a complex grammar to be organised into smaller pieces (external referenced rules) that are like hyperlinks in a common HTML page.

This is a very powerful feature because it adds many potential uses in a very simple and straightforward way. A grammar developer can organize a grammar into a fixed part, which references external URI for grammar rules for the dynamically changing parts of the grammars, i.e. list of names in an address book or list of movie titles, so that only when the grammar is used or cached, will the actual values be fetched, so providing the simplest way of updating a grammar online. Another possible use is to create small reusable grammars that are referenced within larger grammars, so saving development time and enforcing the tuning of the grammar, e.g. if a bug is fixed in a reusable grammar then all the referencing ones can benefit from this.

This feature has the potential to change the organization and the development of ASR resources, to save investments and enlarge the performance of speech applications.

Other features are described in the SRGS specification ([3]) and they allow a flexible encoding of a speech grammar, for instance the repetitions are allowed in a very flexible way, and special rules, especially for skipping *garbage* words that do not need to be completely recognized.

The presence of metadata in the grammar may allow, in future, the searchability of grammar resources on the Web, like Semantic Web.

### ***Is it possible to change the grammar format?***

Yes, even if a grammar is written in ABNF, it can be transformed into XML. From a XML SRGS grammar, a simple XSL can transform it into many other common speech grammar formats. This means that is very easy to transform a grammar from SRGS into other formats, but viceversa is not so easy. The SRGS formalism, therefore, is the most versatile one.

### ***What does Semantic Interpretation mean?***

Speech recognition grammars are divided in two distinct parts, the *syntactical part*, that limits the sequence of words a speaker can pronounce for recognition, and the *semantic part*, that helps to generate results from speech recognition. The Semantic Interpretation is this second part of the grammar.

There is a very simple usage of Semantic Interpretation, which allows aliases for recognized words or reduces a recognized word into a codified form, e.g. the US state into a short form - "Massachusetts" is returned to application as "MA". Other uses are more complex, and they may include the transformation of some part of the recognized data into more structured results. This transformation may also include a *validation* of the recognized results, e.g. Feb 29, 2006 is not a valid date because the year 2006 is not a leap year and does not include Feb 29.

### ***Why a standard for Semantic Interpretation?***

A standard grammar formalism like SRGS is not enough to allow a full exploitation of the standards in the grammar development. An important piece is missing, so the result is a partially complete standardization of the ASR grammars, that limits their real portability.

The W3C Voice Browser WG is also working on Semantic Interpretation. The "Semantic Interpretation for Speech Recognition" (SISR) specification ([4]), even though it does not yet have W3C Recommendation, is at a very advanced stage of standardization.

The SISR specification allows the two uses of Semantic Interpretation mentioned above. The simple substitution is called "Literal semantics", and it is restricted to returning a string of characters as semantic results, while the more complex one is based on a well-known scripting language, JavaScript, now ECMA-327 standard [5].

## ***Why ECMA-327? Is it JavaScript too?***

ECMA-327 is a restricted version of ECMA-262 ([6]), the formal name of JavaScript. The restrictions are all motivated by the reduction of the computational cost of the interpretation of the script. This to allow a very powerful Semantic Interpretation, but in the meantime to allow an efficient implementation of the Semantic Interpretation processors.

## ***Are there computational efficiency issues related to Semantic Interpretation?***

Yes, because the speech recognition process is a time consuming activity and there could be a multiplicity of results to be generated after the recognition. The so-called N-best recognition results: in many applications it is useful to let the ASR produce not only the best recognition result, but also the second-best results, the third-best results and so-on. This means that the Semantic Interpretation processing might be repeated many times after each single recognition turn.

The consequences are that the very powerful scripting language needs to be used without assumptions that the inefficiency will not impact the speech application, but this does not undermine the value of the Semantic Interpretation itself. It is only a caution that the developers need to keep in mind.

## ***Will SISR impact on grammar development?***

Yes, because a SRGS grammar with SISR will be very powerful, to include more advanced results than all the other speech recognition grammar in use today. This will further push in the direction of creating powerful grammar resources with very well defined and useful results.

## ***And on VUI design?***

It is not an impact on the VUI design itself, but on the architecture of a speech application that can be simplified if all the possible levels of processing are used. Some processing may move from the application server into the speech recognition grammar to obtain more reliable speech recognition results.

This aspect will also promote an improvement of a new generation of speech applications, their performance and also the maintainability of large speech applications might take advantage of these aspects.

## ***What about the good old DTMF grammars?***

The fact is that today many applications are still based on DTMF for input, or they allow a double input: either speech or DTMF, as a fallback. The issue of DTMF grammars is not a left-over from the early days of the speech application age. They will remain active even in the future.

The SRGS grammar formalism has been created to cover both the speech recognition aspects and also the much more limited aspects of grammars for DTMF usage. Exactly the same grammar, with all the benefits above mentioned, can be used in DTMF only applications.

The SISR specification presence can therefore be assured for DTMF grammars too. This is not insignificant, because all the benefits and power of SISR can be applied to both speech and DTMF grammars. This will allow a speech application to receive exactly the same results from either DTMF or speech, and to reduce the cost and complexity of the application itself.

## ***Are there other relevant standards on the way?***

Yes, for instance W3C Voice Browser WG is working on the standardization of a Pronunciation Lexicon Specification (PLS) [7], which will impact on both speech recognition grammar and speech synthesis. The PLS will allow the creation of a list of words expressed in a phonetic alphabet. This is useful in many areas, for instance for proper names, addresses, etc.

A further W3C specification is the Extensible Multimodal Markup Annotation (EMMA) [8], the focus of which is to produce speech recognition results in a rich XML format that is suitable even for multimodal application.

Last but not least, IETF is working on the creation of standard protocol called Media Resource Control Protocol version 2 (MRCPv2) [9], which is a major candidate for being very widely adopted in the integration of speech servers into speech platforms. The speech recognition and results standards have a direct impact on it too.

The aim of this paper is to highlight the major issues regarding standards for speech recognition. The main points have been mentioned above, but Loquendo strongly believes that the application of standards will become an important driving factor in scaling up the size and impact of speech application over the coming years.

## ***Acronyms:***

ABNF	Augmented Backus-Naur Form
ASR	Automatic Speech Recognition
DTD	Document Type Definition
DTMF	Dual-Tone MultiFrequency
ECMA	European Computer Manufacturers Association, [10]
EMMA	Extensible Multimodal Markup Annotation, [8]
HTML	HyperText Markup Language
IETF	Internet Engineering Task Force, [2]
MRCP	Media Resource Control Protocol, [9]
PLS	Pronunciation Lexicon Specification, [7]
SISR	Semantic Interpretation for Speech Recognition, [4]
SRGS	Speech Recognition Grammar Specification, [3]
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VoiceXML	Voice Extensible Markup Language, [11]
W3C	World Wide Web Consortium, [12]
XSL	eXtensible Stylesheet Language, [13]

## **References:**

- [1] W3C Voice Browser Working Group, web site: <http://www.w3.org/voice/>
- [2] Internet Engineering Task Force, web site: <http://www.ietf.org/>
- [3] Hunt, A., McGlashan S., "Speech Recognition Grammar Specification Version 1.0", W3C Recommendation, March 2004, see: <http://www.w3.org/TR/speech-grammar/>
- [4] van Tichelen, L., Burke D., "Semantic Interpretation for Speech Recognition (SISR) Version 1.0", W3C Last Call Working Draft, November 2004, see: <http://www.w3.org/TR/semantic-interpretation/>
- [5] ECMA-327 Standard "ECMAScript 3<sup>rd</sup> Edition Compact Profile", ECMA, June 2001, see: <http://www.ecma-international.org/publications/standards/Ecma-327.htm>
- [6] ECMAScript Language Specification (ECMA-262), 3<sup>rd</sup> Edition, ECMA, December 1999, see: <http://www.ecma-international.org/publications/standards/Ecma-262.htm>
- [7] Baggia, P., "Pronunciation Lexicon Specification (PLS) Version 1.0", W3C Working Draft, February 2005, see: <http://www.w3.org/TR/pronunciation-lexicon/>
- [8] Johnston, M., Chow W., Dahl D., McCobb G., Raggett D., "EMMA: Extensible MultiModal Annotation markup Language", W3C Last Call Working Draft, September 2005, see: <http://www.w3.org/TR/emma/>
- [9] Shanmugham, S., Burnett D., "Media Resource Control Protocol Version 2 (MRCPv2)", IETF, December 2005, see: <http://www.ietf.org/internet-drafts/draft-ietf-speechsc-mrcpv2-09.txt>
- [10] European Computer Manufacturers Association, web site: <http://www.ecma-international.org>
- [11] Mc Glashan, S., et al., "Voice Extensible Markup Language (VoiceXML) Version 2.0", W3C Recommendation, March 2004, see: <http://www.w3.org/TR/voicexml20/>
- [12] World Wide Web Consortium (W3C), web site: <http://www.w3.org>
- [13] Kay, M., "XSL Transformations (XSLT) Version 2.0", W3C Candidate Recommendation, November 2005, see: <http://www.w3.org/TR/xslt20/>