

Verbio Software Reference

Guía del usuario para Windows



Verbio Technologies, S.L.

Verbio Software Reference: Guía del usuario para Windows



Verbio Technologies, S.L.

publicado Septiembre de 2009

Copyright © 2009 Verbio Technologies, S.L.

Tabla de contenidos

1. Introducción	1
1. ¿Qué es Verbio?	1
2. Uso de esta guía	1
3. Más información	1
2. Inicialización, configuración y licencias	3
1. Prerrequisitos	3
2. Módulos obligatorios	3
2.1. Componentes del servidor (Verbio Server Engine)	3
2.2. Componentes del cliente (Verbio Client Engine)	4
3. Módulos opcionales	5
3.1. Componentes del servidor	5
3.2. Componentes del cliente	6
4. Requisitos hardware	9
4.1. Tamaño del vocabulario	9
4.2. Locutores utilizados	10
4.3. Estrategias de diseño de la arquitectura	11
5. Política de licencias	12
5.1. Licencias de reconocimiento	12
5.2. Licencias de síntesis	13
5.3. Soportes de licencias	13
5.4. Upgrade de licencias	14
6. Puesta en marcha del servidor Verbio	14
6.1. Inicio de Verbio Server Configuration Manager	14
6.2. Especificación de las configuraciones de interés	15
6.3. Especificación de los locutores de interés	16
6.4. Actualización de licencias	17
6.5. Etapa de pruebas	20
6.6. Puesta en marcha del servidor de síntesis y/o reconocimiento	21
6.7. Puesta en marcha del servidor de LOG	21
6.8. Detención y desinstalación del servicio	21
3. Plataformas, herramientas y kits de desarrollo	23
1. Plataformas software y hardware	23
1.1. Integración de la telefonía y los ordenadores (CTI)	23
1.2. Programación de aplicaciones de escritorio	24
2. SDK - Kit de desarrollo de software	25
2.1. Library SDK - Plataforma genérica	25
2.2. Advanced SDK - Plataforma genérica C++	25
2.3. Dialogic SDK - Dialogic	25
2.4. WordSpotting SDK	25
2.5. CT-ADE SDK	26
2.6. Avaya IR SDK	26
2.7. Evox SDK	26
2.8. Microsoft SAPI 4/5 - TTS	26
3. Herramientas del desarrollador	26
3.1. Verbio Grammar Manager	26
3.2. Verbio Read Aloud	26
4. Reconocimiento del habla	27
1. Introducción	27
2. Inicialización	27
3. Gramáticas de reconocimiento	27
3.1. Tipos de gramáticas	27
4. Uso de gramáticas	57
4.1. Crear una gramática	57
4.2. Preparar el vocabulario	57
4.3. Activar una gramática para el reconocimiento	58
5. Reconocer una elocución	58
5.1. Activar el reconocimiento de habla	58
5.2. Formato de las muestras de voz	59

5.3. Barge-in	59
6. Obtener el resultado	59
6.1. Medidas de confianza	59
6.2. Interpretación semántica	59
6.3. Múltiples hipótesis	59
5. Conversión de texto en habla	61
1. Inicialización	61
2. Conceptos básicos	61
3. Marcación de síntesis del habla	61
3.1. Lenguajes de marcación de síntesis del habla	61
4. Diccionarios de excepciones y abreviaturas	65
4.1. Diccionarios de excepciones de Usuario	65
4.2. Diccionarios de abreviaturas de Usuario	65
Índice	67

Lista de tablas

2.1. Configuraciones multilingües	5
2.2. Configuraciones monolingües	5
2.3. Locutores necesarios	6
2.4. Selección del SDK de interés	7
2.5. Peticiones simultáneas en función del tamaño del vocabulario	10
2.6. Peticiones simultáneas de síntesis en tiempo real en función del equipo	10
4.1. Built-in soportadas en cada configuración	33
4.2. Modelos específicos de palabras para español castellano	39
4.3. Modelos específicos de palabras para catalán	41
4.4. Modelos específicos de palabras para euskera	43
4.5. Modelos específicos de palabras para gallego	45
4.6. Modelos específicos de palabras para portugués	46
4.7. Modelos específicos de palabras para portugués brasileño	48
4.8. Modelos específicos de palabras para español argentino	49
4.9. Modelos específicos de palabras para español chileno	50
4.10. Modelos específicos de palabras para español colombiano	51
4.11. Modelos específicos de palabras para español mexicano	52
4.12. Modelos específicos de palabras para español venezolano	53
4.13. Modelos específicos de palabras para francés	54
4.14. Modelos específicos de palabras para inglés americano	56

Lista de ejemplos

4.1. Vocabulario de listas de palabras sencillo	28
4.2. Vocabulario de listas de palabras avanzado	28
4.3. Vocabulario de listas de palabras bilingüe (idioma por defecto español)	29
4.4. Ejemplos de gramáticas ABNF	31

Capítulo 1. Introducción

La manera más natural que tiene un individuo de comunicarse ha sido, es y seguirá siendo la voz, por lo que no es de extrañar el gran esfuerzo que se viene realizando en los últimos años para desarrollar técnicas que permitan extender esta comunicación más allá del ámbito inter-personal. Actualmente, se han conseguido unos resultados más que aceptables para un gran abanico de aplicaciones, de modo que empiezan a aparecer en el mercado productos que incorporan tecnologías del habla para facilitar, automatizar o hacer más natural la interacción hombre-máquina.

1. ¿Qué es Verbio?

Verbio es un conjunto de librerías y utilidades destinadas a conseguir una rápida y sencilla incorporación de herramientas del habla (reconocimiento y síntesis del habla, verificación de locutor y soluciones relacionadas) en aquellas aplicaciones en las que pueda resultar interesante disponer de una interfaz vocal.

Verbio incorpora, por lo tanto, funcionalidades de reconocimiento del habla y de síntesis del habla, cuyas características principales se detallan en los capítulos Capítulo 4. *Reconocimiento del habla* y Capítulo 5. *Conversión de texto en habla* respectivamente.

Cualquier entorno que disponga de un dispositivo que permita obtener y/o reproducir muestras de audio será susceptible de incorporar las herramientas contenidas en *Verbio*. Es decir, el sistema de reconocimiento requerirá la obtención de las muestras de audio dictadas por el locutor para procesarlas y obtener el resultado de reconocimiento. Por otro lado, el sistema de síntesis de voz requerirá la reproducción de las muestras de audio que haya generado a partir del texto introducido.

2. Uso de esta guía

Esta guía pretende ser el nexo de unión de toda la documentación incluída en *Verbio*. Para ello, trata todos los temas necesarios para una mejor comprensión y óptima utilización de los recursos proporcionados por *Verbio*, haciendo referencia a documentos específicos más precisos cuando es necesario. Por lo tanto, es recomendable su lectura detallada antes de proceder con el desarrollo de las aplicaciones porque contiene información muy importante que puede evitar problemas posteriores o, lo que es más preocupante, el uso ineficaz de los sistemas de reconocimiento y síntesis de voz actuales.

En el Capítulo 2. *Inicialización, configuración y licencias* se muestran los aspectos más importantes a la hora de instalar y configurar el servidor (o servidores) que constituyen el motor de *Verbio*. La arquitectura de funcionamiento recomendada es la *cliente-servidor*, dado que facilita el mantenimiento y la redundancia del sistema. Asimismo, también puede encontrarse en este capítulo información referente a las prestaciones del sistema y su consumo de recursos, aspectos que cobran especial interés a la hora de dimensionar el hardware a utilizar.

En el Capítulo 3. *Plataformas, herramientas y kits de desarrollo* se detallan las herramientas que tiene a su disposición el desarrollador para incorporar soluciones de síntesis y reconocimiento del habla en sus aplicaciones. Es importante encontrar el SDK más adecuado en cada caso, puesto que ello conllevará una mayor simplicidad y rapidez de implementación. *Verbio*, además, incluye un conjunto de herramientas de gran utilidad para el desarrollo, las pruebas y el posterior mantenimiento de las aplicaciones desarrolladas.

En el Capítulo 4. *Reconocimiento del habla* se detallan los aspectos a tener en cuenta a la hora de usar las herramientas de reconocimiento del habla. En particular, se hace especial hincapié en dos aspectos que son cruciales para un óptimo uso del reconocedor: la correcta creación de gramáticas de reconocimiento y la correcta interpretación de los resultados devueltos por el reconocedor.

En el Capítulo 5. *Conversión de texto en habla* se detallan los aspectos a tener en cuenta a la hora de usar las herramientas de síntesis del habla. En particular, se describen aspectos como la modificación puntual del comportamiento del TTS seleccionado y la especificación de abreviaturas y acrónimos a tener en cuenta para una correcta reproducción del texto especificado.

3. Más información

Documentación sobre los diferentes SDK:

- Verbio Software Reference: Referencia de las funciones del Advanced SDK
- Verbio Software Reference: Referencia de las funciones del Library SDK
- Verbio Software Reference: Referencia de las funciones del Dialogic SDK
- Verbio Software Reference: Referencia de las funciones del CT-ADE SDK
- Verbio Software Reference: Referencia de las funciones del Avaya IR SDK
- Verbio Software Reference: Referencia de las funciones del EVOX SDK

Herramientas de desarrollo:

- Verbio Software Reference: Guía del usuario de Grammar Manager
- Verbio Software Reference: Guía del usuario de Read Aloud

Información de contacto:

En caso de detectar errores en la documentación proporcionada con cualquiera de las versiones de Verbio o bien en caso de requerir soporte adicional para el desarrollo de aplicaciones no dude en contactar con el departamento técnico mediante la dirección de correo support@verbio.com.

En caso de requerir información adicional referente a precios, licencias y cualquier otro aspecto de índole comercial, puede contactar con el departamento comercial mediante la dirección de correo info@verbio.com.

Capítulo 2. Inicialización, configuración y licencias

1. Prerrequisitos

Los productos *Verbio* actualmente están disponibles para sistemas operativos Windows en sus variantes 2000/XP/2003/Vista, y para la mayor parte de las distribuciones Linux (consúltenos la disponibilidad en support@verbio.com). Este documento está enfocado a la instalación y configuración de *Verbio* sobre la plataforma Windows. En caso de querer instalar y utilizar *Verbio* sobre Windows, consulte *Verbio Software Reference: Guía del usuario para Linux* (<http://www.verbio.com/webverbio3/html/reference/README-INSTALL-VERBIO.txt>). Antes de proceder a la instalación del software es conveniente verificar que el hardware utilizado reúne los requisitos mínimos indispensables para soportar la carga computacional y de memoria consumidos por los sistemas de reconocimiento y síntesis del habla *Verbio*. En la Sección 4 de este capítulo se detallan los aspectos necesarios para dimensionar un equipo o varios equipos en función de las condiciones de trabajo particulares: tamaño de las gramáticas de reconocimiento, locutores a utilizar, etc.

El uso de los sistemas de reconocimiento del habla, síntesis del habla y de las herramientas que constituyen la variedad de productos englobados bajo el nombre comercial *Verbio* requiere la instalación de un conjunto de módulos distribuidos en instaladores independientes. De este modo, únicamente es necesario descargar e instalar, en el orden adecuado, aquél o aquellos módulos estrictamente necesarios. El objetivo de este documento es guiar a los instaladores para que puedan determinar los módulos que requieren, así como para que puedan proceder a su posterior instalación. Los módulos pueden clasificarse en base a distintos criterios, aunque las principales subdivisiones podrían ser:

- Módulos obligatorios y módulos opcionales
- Módulos de servidor y módulos de cliente
- Módulos de reconocimiento del habla y módulos de síntesis del habla

2. Módulos obligatorios

Los motores de reconocimiento y síntesis del habla *Verbio* están diseñados para funcionar en una arquitectura cliente-servidor. Es decir, se requiere uno o más servicios (*servidores*) que atienden las peticiones de uno o más clientes. Entendemos por cliente toda aplicación que utilice recursos de reconocimiento y/o síntesis del habla. En particular, todas aquellas máquinas que contienen el dispositivo de audio (tarjeta CTI o de sonido). Esta arquitectura permite diseñar un entorno de trabajo fácilmente escalable y con redundancia. En entornos de poca carga computacional o en aquellos que lo requieran, el cliente y el servidor pueden estar funcionando en la misma máquina.

2.1. Componentes del servidor (*Verbio Server Engine*)

Debe instalarse en todas aquellas máquinas destinadas a alojar un *servidor* (sólo se permite un único *servidor* por máquina), ya sea de reconocimiento del habla, de síntesis del habla o de ambos simultáneamente. Para instalar *Verbio Server Engine*, debe ejecutarse el instalador **VerbioEngines.exe**. Una vez seleccionado el directorio de instalación, aparece la ventana de selección de los componentes que deben instalarse. Si en la máquina actual únicamente debe instalarse el servidor, se seleccionará sólo la opción *Componentes del servidor*. En caso de que se desee instalar el cliente y el servidor en la misma máquina, deben seleccionarse los dos componentes.



Selección de Componentes del Servidor y/o Componentes del Cliente.

Una vez instalados los componentes específicos de *Verbio*, el instalador comprueba que en el equipo estén presentes los driver de las llaves de licenciamiento (o mochilas o sentinels o candados) requeridas para la puesta en marcha del servicio. Las llaves son dispositivos hardware que deben conectarse al puerto paralelo o USB de la(s) máquina(s) en la(s) que se aloja(n) el(los) servidor(es). Para obtener las llaves debe contactar con su proveedor habitual. En caso de que los driver no están instalados, se ofrece la posibilidad de hacerlo.



Opción de instalar los driver de las llaves de licenciamiento si no están previamente instalados.

En caso de proceder a la instalación, deben seleccionarse las opciones por defecto.

Atención

En caso de que en el equipo haya insertada alguna llave, es imprescindible retirarla antes de proceder con la instalación de los driver. En caso contrario, podría perderse la información contenida en ellas.

2.2. Componentes del cliente (Verbio Client Engine)

Debe instalarse en todas aquellas máquinas destinadas a alojar uno o varios clientes, independientemente del tipo de transacción que deban realizar.

Para instalar Verbio Client Engine, debe ejecutarse el instalador **VerbioEngines.exe**. Una vez seleccionado el directorio de instalación, aparece la ventana de selección de los componentes que deben instalarse. Si en la máquina actual únicamente debe instalarse el cliente, se seleccionará sólo la opción **Componentes del cliente**. En caso de que se desee instalar el cliente y el servidor en la misma máquina, deben seleccionarse los dos componentes.

3. Módulos opcionales

3.1. Componentes del servidor

Una vez instalados los módulos obligatorios del servidor según lo comentado en los apartados previos, deberá procederse a instalar aquellos componentes opcionales necesarios. *Verbio* distribuye módulos de reconocimiento del habla (**configuraciones**) y módulos de síntesis del habla (**locutores**). Así, pues, en función de los idiomas que deba soportar el motor de reconocimiento será necesario instalar una o más **configuraciones**. Asimismo, para cada locutor que se desee utilizar, también será necesario instalar su **locutor** correspondiente.

3.1.1. Configuraciones de reconocimiento

Cada **configuración** permite la inclusión de uno o más idiomas en el servidor de reconocimiento del habla en el que se instala. Por lo tanto, únicamente podrán reconocerse locuciones en aquellos idiomas para los que se haya instalado la correspondiente **configuración**. Antes de la utilización de un vocabulario (preparación, activación, reconocimiento, etc.) es necesario activar la **configuración** asociada al idioma de trabajo (mediante la función correspondiente del API de programación), de modo que puede ser necesario alternarla en aquellas aplicaciones multilingües.

Un caso particular lo constituyen las **configuraciones multilingües** que permiten la utilización de vocabularios y/o gramáticas multilingües; es decir, vocabularios y/o gramáticas que contienen entradas en varios idiomas. En estos casos, no es necesario modificar la **configuración** al pasar de un idioma a otro (siempre y cuando todos ellos estén incluidos en la **configuración** seleccionada), sino que es suficiente con indicar al sistema el idioma de trabajo por defecto. El uso de estas **configuraciones** es muy útil en aquellos casos en los que se desconoce el idioma a emplear por los usuarios, especialmente para nombres propios, municipios, etc. en zonas con pluralidad lingüística. Actualmente sólo se tienen disponibles **configuraciones multilingües** de los idiomas oficiales de España. Para mayor información acerca de cómo trabajar con vocabularios y/o gramáticas multilingües, consulte la Sección 3.1.1.1.

En la tabla siguiente se muestran las **configuraciones** que deben instalarse para soportar los distintos idiomas disponibles en *Verbio*, prestando especial atención a los casos en los que se desee disponer de **configuraciones multilingües**. El módulo que debe instalarse en cada caso se compone de **ASRConfiguration** más el contenido de la columna **Instalador**.

Tabla 2.1. Configuraciones multilingües

Español	Catalán	Euskera	Gallego	Instalador
y	n	n	n	es
o	y	n	n	es_ca
o	n	y	n	es_eu
o	n	n	y	es_ga
y/o	y/o	y/o	y/o	es_ca_eu_ga
(y) requerido, (n) no requerido, (o) opcional				

Tabla 2.2. Configuraciones monolingües

Idioma	Instalador
Español argentino	es-ar
Español chileno	es-cl
Español colombiano	es-co
Español mexicano	es-mx
Español venezolano	es-ve
Francés	fr
Inglés americano	en-us
Portugués	pt
Portugués brasileño	pt-br

Atención

Atención, contacte con VERBIO (info@verbio.com) si requiere algún idioma adicional, puesto que se está en fase de desarrollo de nuevas configuraciones de reconocimiento.

3.1.2. Locutores de síntesis

Para disponer de un locutor determinado en el servidor de síntesis, es imprescindible instalar su locutor correspondiente. En función de los idiomas que se deseen soportar, deben instalarse uno o más de los locutores disponibles para cada idioma. Mediante la función correspondiente del API de programación deberá seleccionarse el idioma (y, en ocasiones, el locutor) adecuado para cada caso, en función de las necesidades de la aplicación.

En la tabla siguiente se muestran los locutores disponibles en *Verbio* para cada uno de los idiomas soportados.

Tabla 2.3. Locutores necesarios

Idioma	Sexo	Edad	Nombre	Instalador
Catalán	Hombre	Adulto	Oriol	Catalan-Oriol
Catalán	Mujer	Adulta	Meritxell	Catalan-Meritxell
Español castellano	Hombre	Adulto	Carlos	Spanish-Carlos
Español castellano	Mujer	Adulta	Laura	Spanish-Laura
Español castellano	Mujer	Adulta	Amaya	Spanish-Amaya
Español argentino	Hombre	Adulto	Javier	Argentinian-Javier
Español mexicano	Mujer	Adulta	Celia	Mexican-Celia
Euskera	Mujer	Adulto	Amaia	Basque-Amaia
Francés	Mujer	Adulta	Brigitte	French-Brigitte
Gallego	Hombre	Adulto	Freire	Galician-Freire
Inglés americano	Mujer	Adulta	Jane	US English-Jane
Portugués	Mujer	Adulta	Adriana	Portuguese-Adriana
Portugués brasileño	Mujer	Adulta	Julia	PortugueseBrazilian-Julia

Atención

Atención, contacte con VERBIO (info@verbio.com) si requiere algún idioma adicional, puesto que se está en fase de desarrollo de nuevos locutores de síntesis.

3.2. Componentes del cliente

Dentro de los módulos adicionales para el cliente, encapsulados dentro del instalador **VerbioDeveloper.exe**, se encuentran los siguientes componentes:



Conjunto de componentes (clientes y herramientas) disponibles para el equipo cliente.

3.2.1. Plataformas de desarrollo de aplicaciones (tipos de cliente)

Verbio proporciona un conjunto plataformas de desarrollo para que integradores de múltiples entornos puedan incorporar tecnologías del habla a sus productos. Para cada una de las plataformas disponibles se proporciona:

- **Software Development Kit (SDK):** Conjunto de ficheros destinados a programar nuevas aplicaciones que incorporen reconocimiento y/o síntesis del habla (librerías, header, etc.).
- **Ejemplos:** Códigos de ejemplo de la utilización del SDK.
- **Documentación:** Descripción de las funciones contenidas en el SDK (*Function Reference*).

La decisión del SDK a utilizar depende de un conjunto de factores que a continuación se detallan con la finalidad de que cada integrador encuentre aquél que mejor se adapte a su entorno de programación y a sus conocimientos. La tabla siguiente es un resumen de las particularidades de cada SDK, pudiéndose consultar la Sección 1 para mayor información.

Tabla 2.4. Selección del SDK de interés

SDK	Entorno de programación	Hardware utilizado	Compatibilidad	Herramientas avanzadas	DLL contenedora
Advanced	Microsoft Visual Studio C/C++	Cualquier tarjeta CTI o dispositivo	Incompatible con versiones	Sí	verbiolib.dll

SDK	Entorno de programación	Hardware utilizado	Compatibilidad	Herramientas avanzadas	DLL contenedora
		I/O	anteriores		
Dialogic	Microsoft Visual Studio C/C++ Otros compiladores C/C++	Tarjetas CTI Dialogic	Compatible con Verbio Dialogic (vx_)	No	vxxxlib.dll
Library	Microsoft Visual Studio C/C++ Otros compiladores C/C++	Cualquier tarjeta CTI o dispositivo I/O	Compatible con Verbio Library (vox_)	No	voxlib.dll
WordSpotting	Microsoft Visual Studio C/C++ Otros compiladores C/C++	Cualquier tarjeta CTI o dispositivo I/O	Específica para tareas de WS	Sí	verbiows.dll, verbiowsc.dll
CT ADE	Graphical VOS (Parity) - CT ADE	Tarjetas CTI Dialogic	Compatible con versiones anteriores de IVR	No	ivr.dll

Es posible utilizar todos los SDK de desarrollo desde compiladores que no trabajen en C/C++, aunque la programación requerirá de un trabajo previo consistente en importar todas las funciones contenidas en la DLL del SDK de interés dentro del entorno de programación.

Verbio incorpora la plataforma de desarrollo Advanced para facilitar el uso de los motores de reconocimiento y síntesis del habla, sobretodo de las nuevas funcionalidades incorporadas. El resto de plataformas se han actualizado para poder disfrutar también de las nuevas prestaciones de *Verbio* sin perder la compatibilidad con las versiones anteriores, aunque ello implique un uso algo más complejo que el de la nueva plataforma.

Las herramientas avanzadas (únicamente disponibles en la plataforma de desarrollo Advanced) afectan tanto a la síntesis como al reconocimiento, aunque es en este último en el que más se ha profundizado, especialmente a la hora de procesar los resultados devueltos por el sistema de reconocimiento.

3.2.2. Herramientas de prueba y desarrollo

- **Verbio Grammar Manager:** Herramienta destinada a diseñar y probar las gramáticas de reconocimiento utilizadas en la aplicación. Para más información consulte la documentación asociada *Verbio Software Reference: Guía del usuario de Grammar Manager*.
- **Verbio Read Aloud:** Herramienta destinada a probar los locutores disponibles mediante la especificación de cualquier texto y la manipulación de sus parámetros de funcionamiento básicos. Para más información consulte la documentación asociada *Verbio Software Reference: Guía del usuario de Read Aloud*.

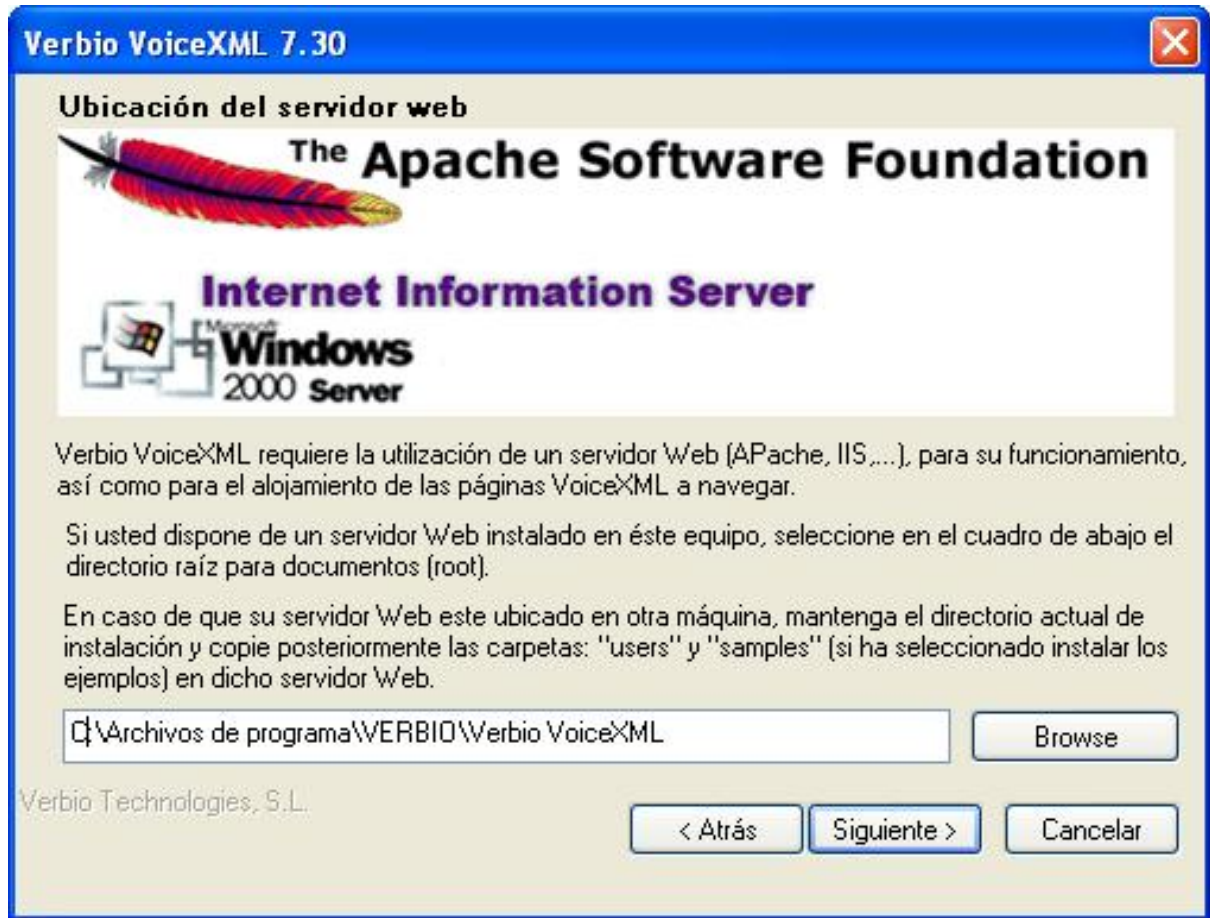
La instalación de las herramientas de prueba y desarrollo no requiere la instalación de ninguno de los tipos de clientes que los acompañan en el instalador, sino que únicamente requieren los componentes obligatorios del cliente. Para más información acerca de cualquiera de estas herramientas, consulte la documentación correspondiente (incluida al instalarlas).

3.2.3. Intérprete de VoiceXML

Dentro de los módulos adicionales para el cliente, *Verbio* también incluye un intérprete basado en el estándar VoiceXML: *VerbioVoiceXML*, contenida dentro del instalador **VerbioVoiceXML.exe**.

Esta herramienta está indicada para todos aquellos desarrolladores e integradores que requieren un entorno de desarrollo sencillo (no requiere de programación a bajo nivel), flexible (no requiere conocer los aspectos técnicos del hardware telefónico a utilizar) y plenamente integrado con las tecnologías del habla de *Verbio*. Para una mayor información, consulte la documentación del producto o la información de soporte en <http://www.verbio.com>.

La instalación únicamente requiere especificar el directorio de instalación (puede ser distinto a aquél en el que se han instalado los componentes obligatorios de *Verbio*) y especificar si se desea instalar los ejemplos y la documentación referentes al producto. Se aconseja utilizar la opción de instalación *completa*, tras lo cual únicamente queda por especificar el directorio del equipo en el que se ha instalado (si está en local) el servidor HTTP (típicamente Apache). En caso de que no se haya instalado el servidor HTTP en el equipo local, se recomienda dejar la opción por defecto.



Proceso de instalación de *Verbio VoiceXML*.

4. Requisitos hardware

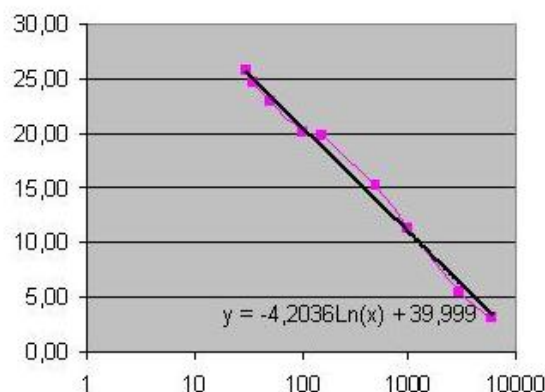
A la hora de determinar el(los) equipo(s) necesario(s) para cada instalación, es conveniente tener en cuenta los aspectos que más influyen en el consumo de recursos hardware. Entre los aspectos más determinantes destacan el tamaño del vocabulario, los locutores utilizados y la distribución de la carga computacional (estrategia de diseño).

4.1. Tamaño del vocabulario

El tamaño de los vocabularios o gramáticas utilizados es, dentro del consumo derivado del uso de tecnologías del habla, el aspecto que más influye en la carga que soporta el procesador. Por lo tanto, a mayor tamaño de la gramática (o gramáticas si se usan varias en paralelo) activa, mayor consumo de CPU. La tabla siguiente puede servir de guía para dimensionar el hardware a utilizar, aunque también hay que tener en cuenta los aspectos que se detallan en la secciones siguientes. Los datos mostrados se han obtenido en un servidor Intel Pentium IV a 2,4 GHz.

Tabla 2.5. Peticiones simultáneas en función del tamaño del vocabulario

Tamaño del vocabulario	Peticiones simultáneas
30	26
35	25
50	23
100	21
150	20
500	15
1000	11
3000	6
6000	3



Peticiones simultáneas en función del tamaño del vocabulario (en palabras) para un Pentium IV a 2,4 GHz.

Por palabras del vocabulario entendemos toda unidad de reconocimiento que forma parte del fichero de vocabulario o de la gramática; es decir, aquella unidad que puede reconocerse de manera aislada. Por ejemplo, en un vocabulario de nombres+apellidos, las palabras del vocabulario estarían constituidas por el nombre más el apellido. Para mayor información acerca de vocabularios y gramáticas, consulte la Sección 3.

Estos resultados se obtienen forzando un tiempo de respuesta inmediato. Es decir, sin permitir retardos entre el final de la locución y la obtención del resultado del reconocimiento. En aplicaciones poco críticas en cuanto a tiempo de respuesta, podrían aumentarse ligeramente el número de puertos simultáneos sin apreciar una degradación de la calidad del servicio.

En algunos casos, sin embargo, es posible desglosar vocabularios y/o gramáticas de gran tamaño en subvocabularios y/o subgramáticas que, con las mismas prestaciones, consumen menos recursos.

En entornos de múltiples canales/puertos, el consumo de memoria también puede ser elevado en algunos casos, puesto que todos los canales tienen cargado en memoria las gramáticas de reconocimiento que deban emplear en el siguiente proceso de reconocimiento. En estos casos, es importante tener activadas las gramáticas únicamente cuando el proceso de reconocimiento deba ser inminente. A título de ejemplo, una gramática que ocupe 4 MBytes en memoria puede ocupar 240 MBytes en entornos de trabajo de 2 primarios (60 canales) si todos los canales la cargan al unísono.

4.2. Locutores utilizados

El uso de síntesis del habla consume una considerable cantidad de memoria RAM del sistema, de modo que ésta deberá adecuarse al tipo y cantidad de locutores utilizados. *Verbio* permite almacenar los datos del sintetizador básicamente todos en disco, de modo que se reduce la cantidad de memoria necesaria, aunque a costa de un tiempo de respuesta ligeramente superior (sobre todo en entornos sobrecargados). Por lo tanto, únicamente se aconseja utilizar esta estrategia en aquellos entornos que requieran pocas transacciones de síntesis simultáneas (del orden de 5) y que tengan limitaciones insalvables de memoria. El consumo de memoria debido a la activación de locutores es elevado, aunque permanece prácticamente constante durante todo el periodo de utilización, independientemente de las peticiones que se le formulen. De este modo, la memoria necesaria viene condicionada por la cantidad de locutores a utilizar y no por la cantidad de peticiones simultáneas que deba ser capaz de procesar el sintetizador. Como punto de partida, para un uso estándar de 2 locutores, es aconsejable un mínimo de 512 MBytes de memoria RAM (preferiblemente de acceso rápido tipo DDRAM). Respecto a la carga computacional, se pueden tener en cuenta las siguientes mediciones:

Tabla 2.6. Peticiones simultáneas de síntesis en tiempo real en función del equipo

Prestaciones máquina	Peticiones simultáneas
Intel Pentium III a 1 GHz	17
Intel Pentium IV a 2,4 GHz	27

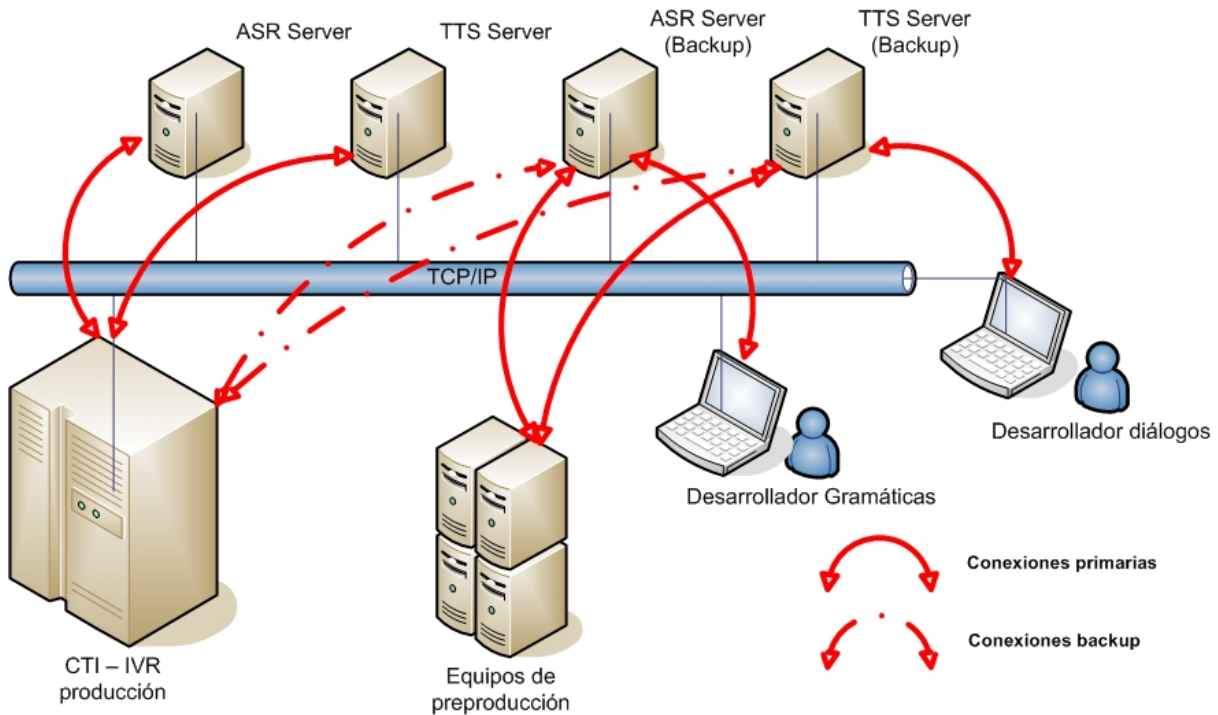
Al igual que para el caso de reconocimiento del habla, estos resultados también se han obtenido forzando un tiempo de respuesta inmediato entre la petición y el inicio de la síntesis. En entornos en los que el tiempo de respuesta no sea crítico, el número de peticiones simultáneas de síntesis puede ser ligeramente superiores a las que aparecen en la tabla sin apreciar una degradación de la calidad del servicio.

4.3. Estrategias de diseño de la arquitectura

Los sistemas del habla desarrollados sobre *Verbio* siguen una estrategia de comunicación cliente-servidor, de modo que en un mismo entorno de trabajo pueden coexistir varios servidores (todos ellos en máquinas distintas) y varios clientes (éstos sí pueden compartir máquina). Este escenario y la posibilidad de cada cliente (concretamente, de cada una de sus líneas por separado) de conectarse a un servidor distinto, permite distribuir la carga computacional entre todos los servidores presentes en el sistema. Con esto se consigue:

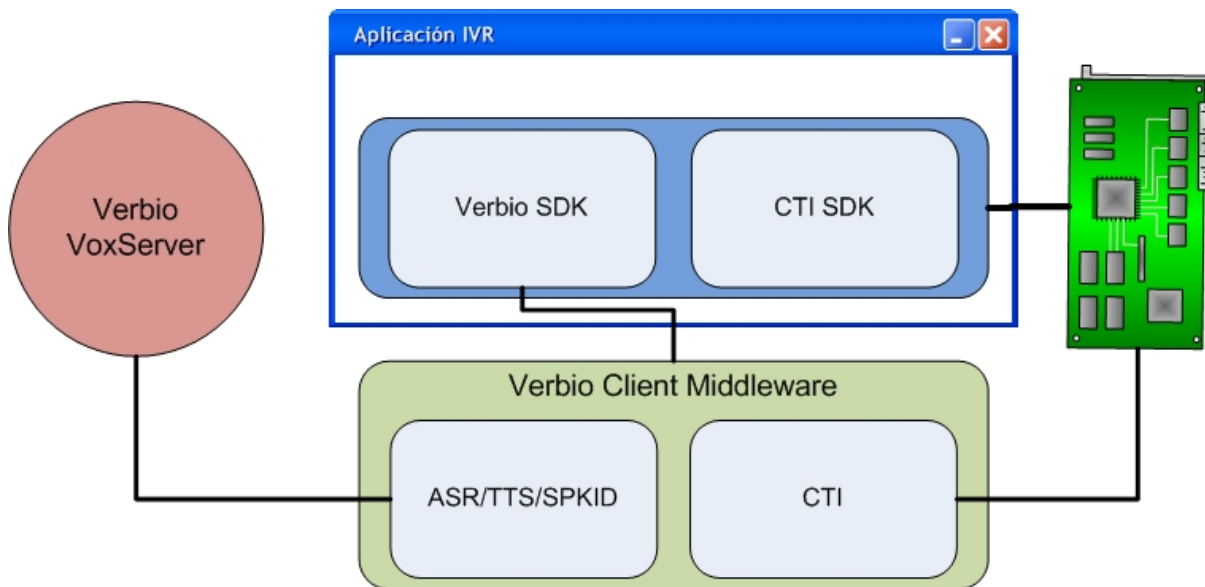
- Reutilización y máximo aprovechamiento de los recursos disponibles: no es necesario deshacerse de equipos antiguos si se aumenta el tamaño de los vocabularios o si el sintetizador se actualiza por otro que consume más memoria o si aumenta el número de líneas atendidas. En cualquiera de estos casos, es suficiente con añadir algún otro servidor al entorno, de modo que redistribuyendo la carga de nuevo entre todos los servidores no sea necesario reemplazarlos todos por máquinas de prestaciones superiores.
- Implementación de estrategias de respaldo (backup): no únicamente es posible que cada línea (de cada cliente) se conecte a un servidor distinto, sino que además es posible indicar servidores de respaldo para que, en caso de caída de algún servidor, las líneas que lo estuvieran utilizando hasta el momento se conecten automáticamente a algún otro de los servidores disponibles. Los servidores de respaldo pueden ser equipos dedicados exclusivamente a esta función o bien ser otros servidores del sistema ya en uso (en este caso, la caída de uno de ellos podría implicar, según el dimensionado, una cierta sobrecarga de los restantes, lo que podría redundar en pequeños retardos, la mayoría de las veces imperceptibles).

En la siguiente figura se muestra un ejemplo de arquitectura cliente-servidor en la que cada cliente (unos en producción y otros en preproducción o desarrollo) se conectan a sus respectivos servidores de síntesis y de reconocimiento. A su vez, estos clientes están configurados de manera que, en caso de pérdida de conexión con su servidor, desvíen sus peticiones hacia el servidor del otro cliente. Esta estrategia permite que ambos clientes sigan operativos en caso de que uno de los servidores caiga. Evidentemente, en caso de que el servidor operativo estuviera ya muy cargado, la sobrecarga derivada de las peticiones del cliente adicional puede provocar ciertos retardos (en momentos pico) en los sistemas, aunque no deja de ser una circunstancia preferible a la de quedarse sin servicio en alguno de los clientes.



Arquitectura cliente-servidor con redundancia a nivel de servidores.

En sistemas que por simplicidad o costes no requieran redundancia ni deban utilizar más de un servidor, *Verbio* permite instalar el cliente (o clientes) y el servidor en la misma máquina, tal y como se muestra en la siguiente figura. Además, sirve la figura para mostrar los módulos internos típicos que suelen estar presentes en las aplicaciones clientes.



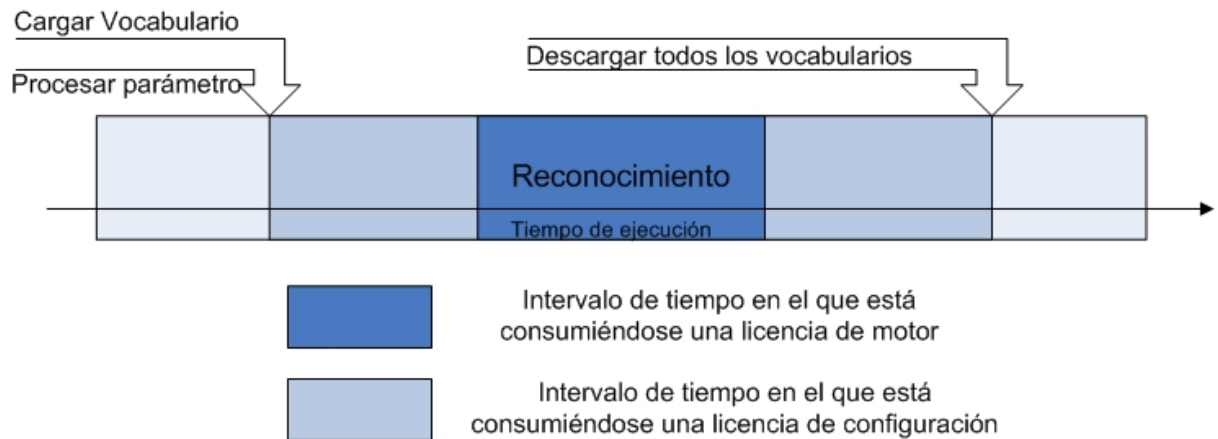
Arquitectura cliente-servidor en entornos de un único servidor.

5. Política de licencias

Verbio dispone de licencias diferenciadas para reconocimiento del habla y para síntesis del habla y, dentro de estos dos grupos, licencias distintas para cada uno de los idiomas soportados. De este modo, sólo es necesario adquirir las licencias necesarias en cada caso, pudiéndose actualizar o incrementar el número de las mismas a medida que se incrementen las necesidades del sistema.

5.1. Licencias de reconocimiento

Las licencias de reconocimiento del habla se dividen en dos grandes grupos: licencia de motor y licencia de configuración. La licencia de motor se consume únicamente durante el proceso de reconocimiento; es decir, mientras el servidor está procesando las muestras de voz para obtener el resultado de reconocimiento. La licencia de configuración está asociada a cada una de las configuraciones disponibles, de modo que para poder utilizar una configuración, es imprescindible disponer de su correspondiente licencia. Las licencias de configuración se consumen mientras haya algún vocabulario cargado asociado a dicha configuración o a partir del momento de la consulta/establecimiento de alguno de los parámetros que rigen el comportamiento del motor de reconocimiento (consulte la documentación técnica para conocer qué parámetros afectan al consumo de licencias). La licencia de configuración se libera cuando se descargan todos los vocabularios. En el momento del reconocimiento, se estará consumiendo, forzosamente, una licencia de motor y una licencia, como mínimo, de configuración (la asociada al(los) vocabulario(s) activo(s)).



Intervalos de tiempo de uso de los distintos tipos de licencias (motor y configuración)

Debido a que el intervalo de tiempo en el que se consumen licencias de configuración es mayor que aquél en el que se consumen licencias de motor, es muy probable que el número de licencias de configuración deba ser superior al de motor si la codificación de la aplicación no ha tenido en cuenta este aspecto.

En caso de reconocimiento en múltiples idiomas (no contenidos en una misma configuración), será necesario disponer de una licencia de motor y de una licencia de cada una de las configuraciones que intervengan en el reconocimiento.

5.2. Licencias de síntesis

Las licencias de síntesis del habla, al igual que las de reconocimiento, se dividen en dos grandes grupos: licencia de motor y licencia de idioma. Ambas licencias se consumen durante el proceso de síntesis; es decir, mientras el motor de síntesis del habla genera y manda las muestras al cliente. Para que el proceso de síntesis se lleve a cabo, es necesario disponer de una licencia de motor y de una licencia de cada uno de los idiomas que estén presentes en el texto a sintetizar. En caso de que falte alguna de ellas, no se sintetizará ningún fragmento del texto.

5.3. Soportes de licencias

Las licencias pueden facilitarse en formato software (fichero) o hardware: mochila (sentinel) de puerto paralelo o USB. La ventaja del formato fichero es la inmediatez de la entrega (se envían por e-mail) y el menor coste (no hay que pagar el suplemento derivado del coste del hardware, aranceles, etc). El principal inconveniente es que dependen de un número de serie cuyo valor es intrínseco a la máquina. Esto impide que la licencia sea operativa en otros equipos (no es portable) o incluso en el mismo equipo después de reinstalaciones del sistema operativo, algunos cambios de hardware, etc. En estos casos es imprescindible adquirir una nueva licencia, puesto que no hay manera de garantizar una migración no fraudulenta. Precisamente las licencias basadas en mochila solucionan este problema al ser independientes del número de serie. Por lo tanto, son reutilizables en cualquier máquina. Los inconvenientes de sobrecoste y falta de disponibilidad inmediata pueden solucionarse adquiriendo las mochilas en el lugar de destino y programándolas mediante Verbio Server Configuration Manager con la ayuda de un fichero de programación que puede ser enviado por e-mail (al igual que se hacía con los ficheros de licencia). Las mochilas utilizadas actualmente son las Rainbow Sentinel Superpro (<http://www.rainbow.com/products/sentinel/superpro.asp>) en su versión puerto paralelo y USB.

5.4. Upgrade de licencias

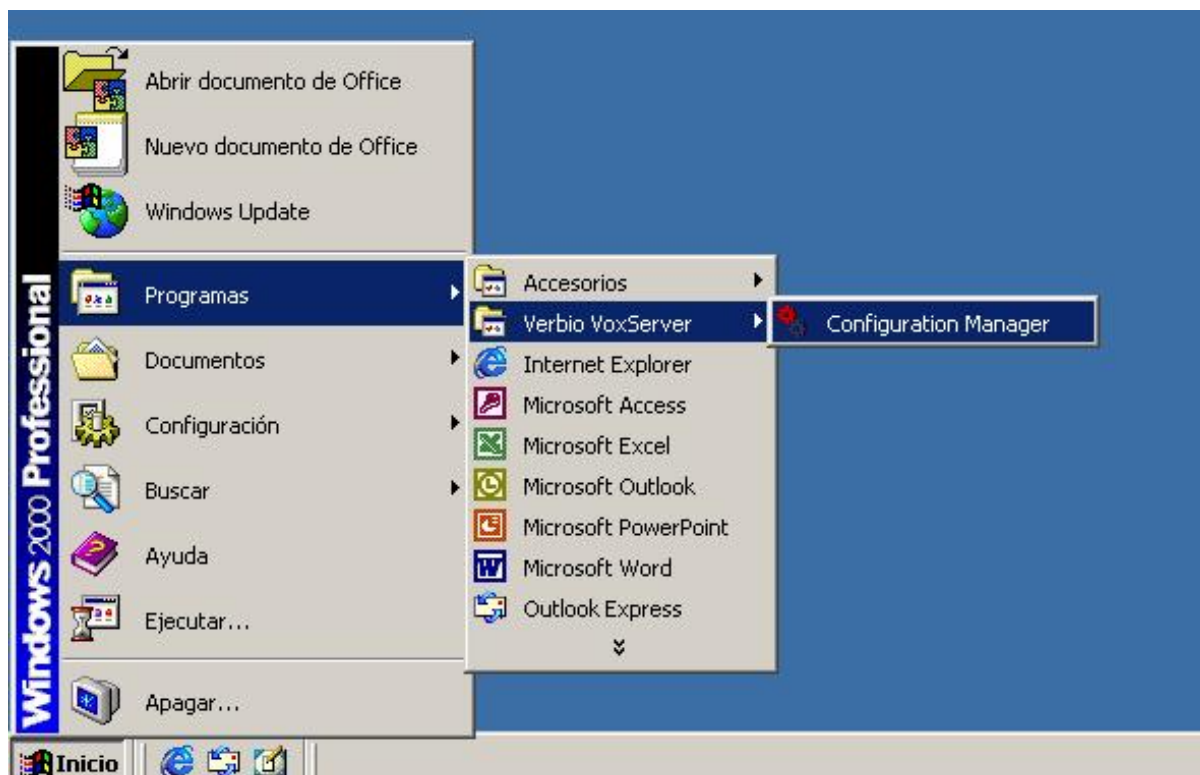
A diferencia de las versiones anteriores de *Verbio 7* (IberVox 6.41 y anteriores), a partir de la versión 7 existen dos tipos de licencias, como se ha comentado en las secciones anteriores. Este factor hay que tenerlo en cuenta a la hora de solicitar upgrade de las versiones anteriores, puesto que para garantizar la compatibilidad total habrá que adquirir las licencias necesarias de cada tipo. A grandes rasgos, las claves para actualizar adecuadamente la versión son las siguientes:

- Hay que adquirir tantas licencias de motor (de ASR y/o TTS) como licencias se tuvieran de las versiones previas.
- Hay que adquirir tantas licencias de idioma TTS como licencias se tuvieran de las versiones previas (se consumen al mismo tiempo que las de motor).
- La cantidad de licencias de configuración ASR a adquirir dependerá de la estrategia de programación seguida durante el desarrollo de la aplicación. En caso de cargar los vocabularios justo antes de lanzar el reconocedor y descargarlos todos al finalizar el proceso de reconocimiento, el número de licencias de configuración coincidiría con las de motor. Cuanto más espaciado esté el tiempo de carga y descarga (o la manipulación de parámetros), más tiempo estará ocupada la licencia de configuración, por lo que es más probable que el sistema se encuentre con todas las licencias ocupadas. En el caso extremo de cargar los vocabularios al inicio de la aplicación y no descargarlos hasta el final, se requerirán tantas licencias de configuración como canales abiertos.

6. Puesta en marcha del servidor Verbio

6.1. Inicio de Verbio Server Configuration Manager

Una vez instalados los módulos obligatorios del servidor y aquellos módulos opcionales necesarios, puede procederse a la puesta en marcha del servidor de reconocimiento y/o síntesis del habla *Verbio*. Para ello, debe usarse *Verbio Server Configuration Manager*.



Arranque del configurador de *Verbio*, *Verbio Server Configuration Manager*

Las operaciones que pueden realizarse desde *Verbio Server Configuration Manager* son las siguientes:

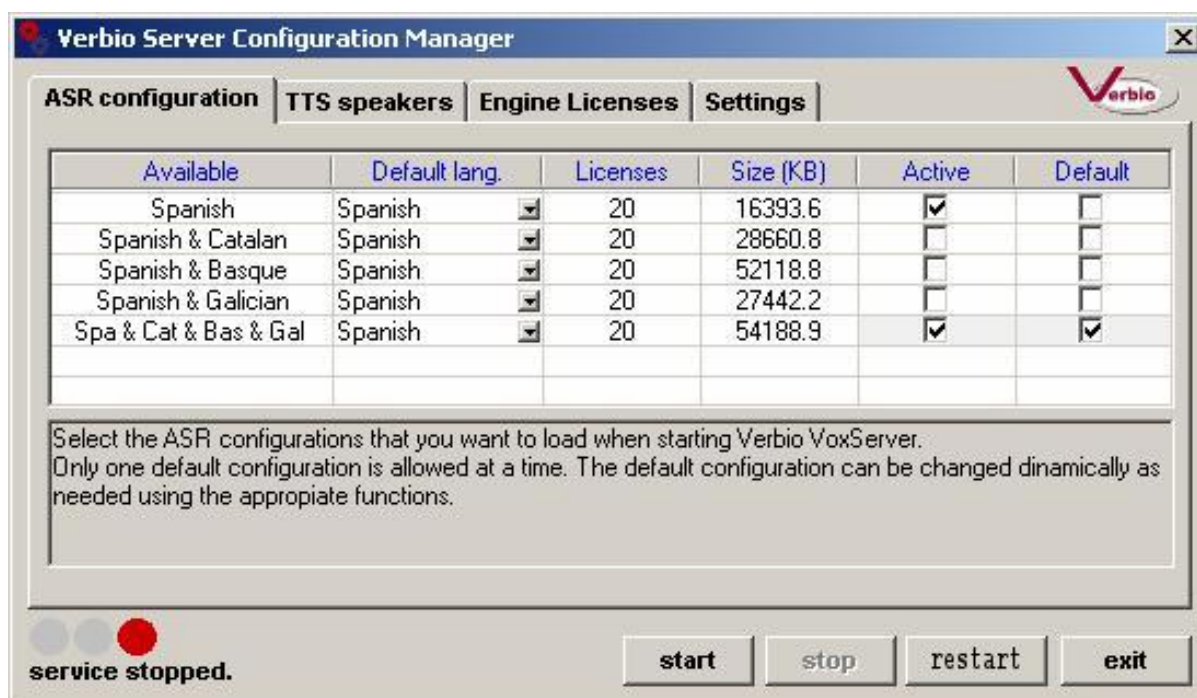
- Arrancar y parar el servidor de reconocimiento y/o síntesis del habla *Verbio*.
- Especificar las configuraciones de reconocimiento que se utilizarán, indicando el idioma por defecto para cada una de ellas así como la configuración por defecto. La configuración por defecto es la que se utilizará para la preparación de los vocabularios y gramáticas si no se indica una configuración diferente. Véase la Sección 6.2.
- Especificar los *locutores* que se utilizarán, así como el *locutor* por defecto para cada idioma. El *locutor* por defecto es el que se utilizará en el proceso de síntesis siempre y cuando no se haya especificado ningún otro locutor y/o idioma previamente. Véase la Sección 6.3.
- Especificar si los *locutores* se cargarán en memoria o bien si accederán directamente a disco. En la sección Sección 4.3 se detallan los aspectos concernientes a esta elección. Véase la Sección 6.3.
- Reprogramar las licencias disonibles remotamente. Véase la Sección 6.4.
- Determinar si el servicio de reconocimiento y/o síntesis del habla *Verbio* debe arrancar automáticamente al iniciarse el equipo y la frecuencia de trabajo del motor de síntesis. Asimismo, se pueden modificar los puertos que utilizan los servicios *Verbio* y *Verbio Logging* para evitar incompatibilidades con servicios que utilicen los mismos puertos o bien para evitar la acción de cortafuegos. Véase la Sección 6.6. Cualquiera de estas modificaciones requiere reiniciar el servicio a continuación para que los cambios tengan efecto.
- Activar el periodo de tuning y especificar el directorio en el que se almacenará toda la información de este periodo. Véase la Sección 6.5
- Activar o detener el servicio de log así como especificar el nivel de detalles que se quieren obtener. Véase la Sección 6.7
- Especificar la frecuencia de trabajo del motor de síntesis de voz 8KHz/16KHz, así como indicar si debe mandarse información del fonema que se está sintetizando en cada instante para sincronismo labial de avatares.

Asimismo, *Verbio Server Configuration Manager* permite visualizar la siguiente información:

- Las configuraciones instaladas en el equipo, así como el número de licencias disponibles para cada una de ellas y el tamaño en disco que ocupan.
- Los locutores disponibles en el sistema, así como sus características, las licencias disponibles para cada uno de ellos y el tamaño que ocupan en caso de ser activados.
- Las licencias disponibles para motores de reconocimiento, síntesis y utilización SAPI.
- El número de serie del servidor y la versión instalada. También se muestra el número de serie de la llave (*sentinel* o *candado*) que contiene las licencias si está insertada en la ranura. En caso contrario, el valor es *-1*.

Una vez arrancado *Verbio Server Configuration Manager*, el primer paso consiste en verificar que se dispone de licencias para todas aquellas configuraciones (pestaña *ASR configuration*) y/o locutores (pestaña *TTS Speakers*) que desean utilizarse. En caso contrario, debe consultarse el número de serie del servidor o de la llave (pestaña *Settings*) y contactar con el proveedor habitual para que éste proporcione el fichero de *upgrade* adecuado. Si no se dispone de licencias, puede probarse el sistema en modo de evaluación en periodos de 60 minutos. Para más información acerca de cómo funciona el sistema de licenciamiento consulte la Sección 6.4 o bien póngase en contacto con su proveedor habitual.

6.2. Especificación de las configuraciones de interés



Especificación y configuración de las configuraciones de reconocimiento de interés.

En caso de querer utilizar reconocimiento del habla, y tras haber verificado que se disponen de licencias tanto de motor como de las configuraciones necesarias (véase el apartado anterior y la Sección 6.4), es necesario indicar al servidor qué configuraciones debe arrancar, en qué idioma (si la configuración es multilingüe) y cuál debe ser la configuración por defecto. La configuración por defecto es la que se utilizará para la preparación de los vocabularios y gramáticas si no se indica una configuración diferente mediante alguna de las funciones de los SDK.

Debe seleccionarse la casilla de la columna Active en todas aquellas configuraciones que deban estar disponibles en el servidor una vez esté operativo. En la Sección 3.1.1 puede obtenerse el listado de las configuraciones necesarias según los requisitos de las aplicaciones (clientes) que se comuniquen con el servidor.

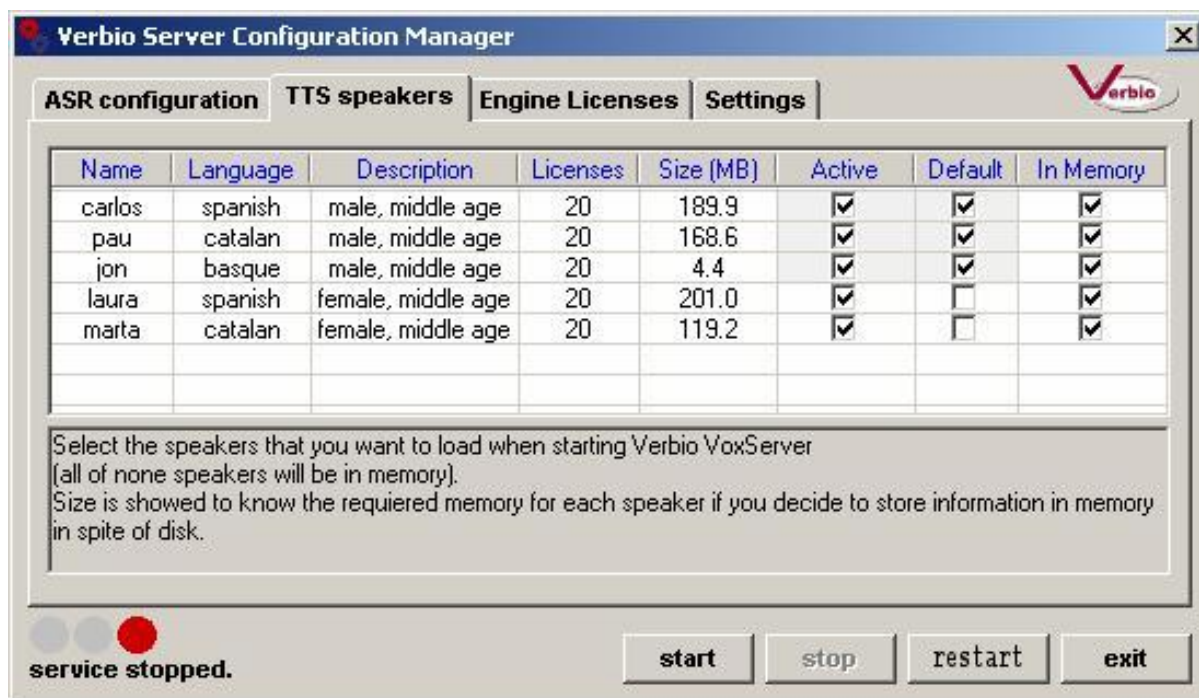
Debe seleccionarse una única configuración por defecto de entre todas las configuraciones activas (por defecto, se selecciona la primera configuración activada) mediante la selección de la casilla correspondiente en la columna Default.

El último paso consiste en seleccionar el idioma por defecto para todas aquellas configuraciones activas multilingües. El idioma por defecto es aquél que se utilizará en la preparación de los vocabularios y gramáticas cuando se utilice su configuración asociada y no se haya indicado lo contrario (por código).

Atención

Es imprescindible detener y volver a arrancar el servicio para que los cambios en la configuración tengan efecto mediante el botón Restart.

6.3. Especificación de los locutores de interés



Especificación y configuración de los locutores de síntesis de interés.

En caso de querer utilizar síntesis del habla y tras haber verificado que se disponen de licencias tanto de motor como de los locutores deseados (véase el apartado anterior y la Sección 6.4), es necesario indicar al servidor qué locutores debe cargar, así como los locutores por defecto para cada idioma.

Debe seleccionarse la casilla de la columna Active de todos aquellos locutores que deban estar disponibles en el servidor una vez esté operativo.

Para cada uno de los idiomas de trabajo debe seleccionarse un locutor por defecto, que será el utilizado en aquellas peticiones en las que no se haya indicado específicamente otro locutor para el idioma afectado. Por defecto se selecciona la casilla Default del primer locutor activado para cada idioma, aunque puede modificarse esta selección posteriormente.

Finalmente, puede optarse por acceder a disco o a memoria (esta medida afecta a todo el conjunto de locutores seleccionados) durante los procesos de síntesis. En la Sección 4.2 se detallan los aspectos a tener en cuenta a la hora de decidirse entre una u otra opción. Si desea accederse a memoria (mayor velocidad aunque mayor consumo de memoria) durante el proceso de síntesis debe seleccionarse la casilla In Memory de cualquiera de los locutores activados. Deben deseleccionarse estas casillas en caso contrario.

También es posible indicar la frecuencia de trabajo del servidor de síntesis. Todos los locutores Verbio permiten trabajar a 8 KHz (para aplicaciones telefónicas) y a 16 KHz (para aplicaciones desktop). Una vez instalados todos los locutores (todos ellos de la misma frecuencia), deberá seleccionarse la deseada mediante el desplegable situado en la pestaña Settings. Es importante tener en cuenta que no puede seleccionarse la frecuencia de trabajo de manera individual para cada locutor.

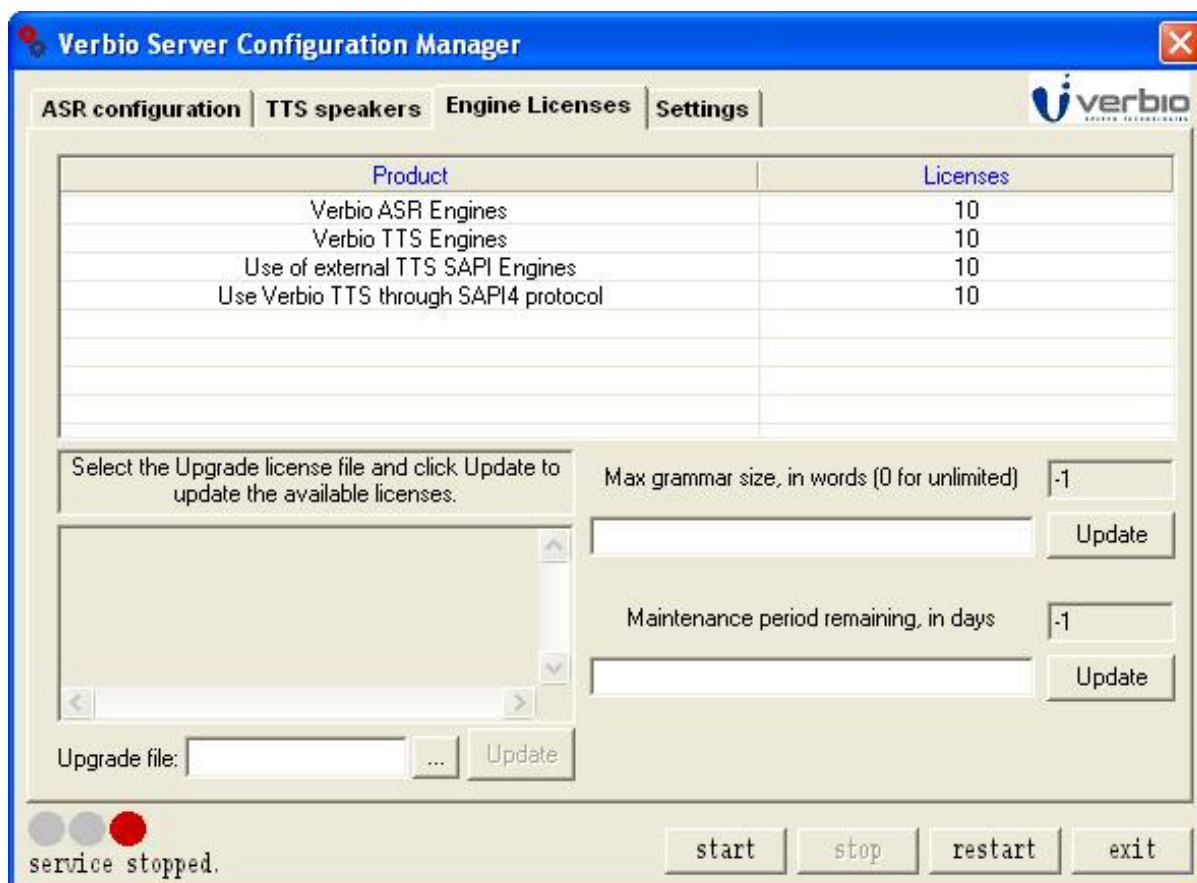
Atención

En caso de que la voz sintética suene distorsionada, posiblemente no se habrá seleccionado correctamente la frecuencia de trabajo en la pestaña Settings. Revise que la frecuencia de trabajo se corresponde con la instalada para todos los locutores.

Atención

Es imprescindible detener y volver a arrancar el servicio para que los cambios en la configuración tengan efecto mediante el botón Restart.

6.4. Actualización de licencias



Consulta y actualización de las licencias disponibles.

El uso de los sistemas de síntesis y reconocimiento del habla de *Verbio* requiere, en función de cada caso, la adquisición de varios tipos de licencias, de modo que se adapten lo mejor posible al uso que se vaya a hacer de ellos.

- *Licencias de motor ASR*: Es necesario disponer de tantas licencias de motor de reconocimiento como peticiones simultáneas de reconocimiento deseen poder atenderse en el servidor, independientemente de las configuraciones utilizadas. La cantidad de licencias de reconocimiento disponibles se muestra en la pestaña *Engine Licenses*.
- *Licencias de motor TTS*: Es necesario disponer de tantas licencias de motor de síntesis como peticiones simultáneas de síntesis deseen poder atenderse en el servidor, independientemente de los idiomas utilizados. La cantidad de licencias de síntesis disponibles se muestra en la pestaña *Engine Licenses*.
- *Licencias de acceso a un SAPI externo*: Es necesario disponer de tantas licencias de motor SAPI externo como peticiones simultáneas deseen poder realizarse desde *Verbio* hacia un locutor compatible SAPI de otro proveedor de tecnologías del habla. La cantidad de licencias de SAPI externo disponibles se muestra en la pestaña *Engine Licenses*.
- *Licencias de acceso vía SAPI*: Es necesario disponer de tantas licencias de motor de acceso SAPI como peticiones simultáneas deseen poder realizarse desde una aplicación compatible SAPI hacia los locutores de *Verbio* utilizando este protocolo. La cantidad de licencias de acceso SAPI disponibles se muestra en la pestaña *Engine Licenses*.
- *Licencias específicas para cada configuración (reconocimiento)*: En función de los idiomas que se deseen utilizar en reconocimiento es necesario disponer de las suficientes licencias (tantas como peticiones simultáneas se deban procesar) de la(s) configuración(es) que los engloba. Lógicamente, el número de licencias de una configuración no debe ser superior al número de licencias de motor ASR disponibles (estarían infrautilizadas). La cantidad de licencias disponibles para cada configuración se muestra en la pestaña *ASR configuration*.

- *Licencias específicas para cada idioma (síntesis)*: En función de los idiomas que se deseen utilizar en síntesis es necesario disponer de las suficientes licencias (tantas como peticiones simultáneas se deban procesar) de dichos idiomas (la licencia para un idioma permite utilizar cualquier locutor de ese idioma). Lógicamente, el número de licencias de un idioma no debe ser superior al número de licencias de motor TTS disponibles (estarían infrutilizadas). La cantidad de licencias disponibles para cada locutor se muestra en la pestaña TTS Speakers.
- *Licencias de prestaciones limitadas (TIER)*: Para aquellos entornos que no requieran de grandes vocabularios y que estén destinados a segmentos de mercado de baja capacidad (en cuanto a número de peticiones atendidas), VERBIO comercializa una versión restringida de su motor de ASR (con o sin TTS) a un precio menor para acercar sus soluciones a sectores y mercados donde el precio es un aspecto crítico para implantar estas tecnologías. Los principales condicionantes son:
 - Tamaño del vocabulario limitado a N palabras, donde N varía en función de la franja de precios.
 - Número máximo de puertos que pueden instalarse en cada entorno. Sólo se acepta una mochila por equipo, con un máximo de L licencias en ella. No se aceptan licencias en formato fichero.

Para más detalles acerca de los límites aplicados a las versiones TIER de VERBIO, póngase en contacto con commercial@verbio.com.

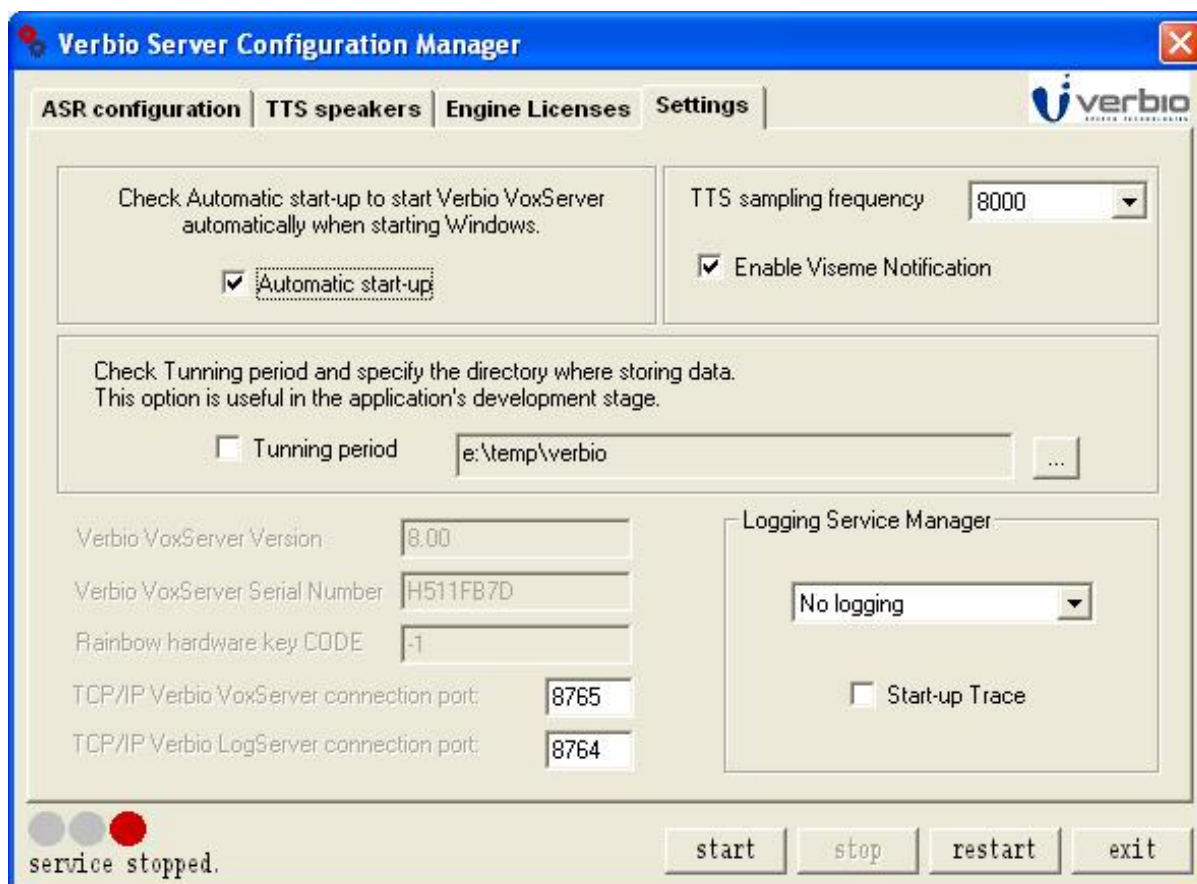
En la pestaña Engine Licenses se muestra el límite (si existe) en el número de palabras que pueden contener los vocabularios y gramáticas. En la casilla Max grammar size, in words aparece el número de palabras máximo, o bien 0 si no hay límite, o bien -1 si no hay ninguna mochila en el equipo. Es posible reprogramar la mochila para limitar o deslimitar el tamaño de las gramáticas mediante un código proporcionado por su proveedor habitual. Dicho código debe copiarse en el campo (Edit box) situado bajo Max grammar size, in words y, posteriormente, pulsar el botón Update. Es necesario reiniciar el servicio Verbio para que los cambios tengan efecto.

- VERBIO permite también alquilar las licencias para un período determinado. En este caso, la licencia también debe ir, obligatoriamente, en formato mochila. Una vez transcurrido el período de alquiler, el servicio deja de funcionar. En caso de querer prorrogar el período de alquiler, debe contactar con su proveedor habitual, quien le hará llegar un código de reprogramación que debe copiar en el campo (Edit box) situado bajo Maintenance period remaining, in days y, posteriormente, pulsar el botón Update. En la casilla Maintenance period remaining, in days aparece el número de días de funcionamiento disponibles, o bien 0 si no hay límite, o bien -1 si no hay ninguna mochila en el equipo.

Las licencias pueden adquirirse/incrementarse solicitando un fichero de Upgrade al proveedor habitual, proporcionando la siguiente información:

- *Tipo de soporte deseado*: Los productos Verbio proporcionan dos soportes distintos para albergar las licencias: fichero de texto y llave hardware. La ventaja del primero es la inmediatez en la entrega (por correo electrónico) y su menor coste debido a no tener que pagar la llave. El principal inconveniente es que la licencia se asocia a un número de serie del producto (o al número de serie de la tarjeta telefónica), de modo que deja de ser válida cuando se cambia de tarjeta telefónica o cuando se reformatea el equipo (el número de serie del producto se obtiene a partir de información del sistema operativo). La llave tiene la ventaja principal de que las licencias son reutilizables en cualquier entorno o equipo en el que se coloque el dispositivo, independientemente del número de serie del producto o de la tarjeta de telefonía utilizada. Además, en caso de avería se sustituye sin problemas por otra (tras la devolución de la defectuosa), cosa que con el soporte fichero no puede realizarse. Los principales inconvenientes son que el tiempo de entrega depende del tiempo de transporte y que el coste se incrementa ligeramente, puesto que incluye los costes de la llave y de los portes. En el caso de optar por llaves, deberá indicarse si se desea una llave de puerto paralelo o bien de puerto USB.
 - *Licencias deseadas*: Debe indicarse el tipo y la cantidad de licencias que se desean obtener (de motor, de configuraciones, etc.). En esta misma sección se han detallado los distintos tipos de licencias existentes. En caso de cualquier duda al respecto, consulte con su proveedor habitual.
 - *Número de serie*: En caso de soporte fichero, debe proporcionarse el número de serie del producto o de la tarjeta CTI. En caso de soporte llave, debe proporcionarse el número de serie de la llave (si se desea reutilizar o reprogramar una llave existente). Ambos números de serie pueden consultarse en la pestaña
-

Settings. En el caso de que no se encuentre una llave en los puertos (USB o paralelo) del equipo, el contenido de la casilla es *-1*.



Números de serie del producto y de la llave necesarios para la obtención de licencias.

En caso de utilizar licencias en soporte llave y desear incrementar el número de licencias, es necesario colocar la llave en la ranura adecuada y solicitar un fichero de Upgrade al proveedor habitual. Este fichero debe seleccionarse en la casilla License file de la pestaña Engine Licenses mediante el explorador de ficheros adjunto. Una vez seleccionado el fichero, en la ventana superior se puede comprobar su contenido (que debe adecuarse a las licencias solicitadas). Tras verificar el contenido del fichero, debe pulsarse el botón **Update** y las licencias quedarán automáticamente actualizadas en la llave.

En caso de utilizar licencias en soporte fichero y desear incrementar el número de licencias, es suficiente con solicitar un nuevo fichero a su proveedor habitual que deberá reemplazar al inicial (el fichero proporcionado ya contendrá la suma de las licencias anteriores y de las solicitadas).

Atención

Es imprescindible detener y volver a arrancar el servicio para que el servidor detecte el incremento de licencias mediante el botón **Restart**.

6.5. Etapa de pruebas

Durante la fase de desarrollo y de pruebas internas de una aplicación es imprescindible disponer de datos que permitan evaluar la calidad del servicio, especialmente del funcionamiento del reconocedor. También es muy aconsejable analizarlos durante el periodo inicial de la fase productiva, puesto que es en la que se podrá tener la certeza del correcto funcionamiento en un entorno real.

Verbio posibilita actualmente el almacenamiento de todas las grabaciones analizadas por el reconocedor, junto con la gramática asociada y los resultados obtenidos para cada una de las locuciones. Esta información se organiza internamente por carpetas para que puedan ser procesadas de forma sencilla con posterioridad. En

breve se proporcionará una nueva herramienta que permitirá un análisis más preciso, sencillo y rápido de los resultados obtenidos, de modo que el desarrollador pueda conocer con la mayor exactitud posible la calidad del servicio y, llegado el caso, la fuente de error más probable en caso de que los resultados no sean los esperados. Esta herramienta, además, permitirá modificar los parámetros del sistema (configuración del motor de reconocimiento, gramáticas, tiempos de guarda, etc.) y volver a evaluar los resultados con el objetivo de encontrar la combinación que mejores resultados ofrecerá con el sistema en producción.

Para activar el registro de toda la información necesaria, debe activarse la casilla `Tuning period` de la pestaña `Settings` del `Verbio Server Configuration Manager`, indicando además el directorio raíz a partir del cual se estructurarán las carpetas con toda la información almacenada. En cuanto finalice la etapa de `tuning`, bastará con desactivar dicha casilla para dejar de almacenar la información.

6.6. Puesta en marcha del servidor de síntesis y/o reconocimiento

Una vez seleccionados las configuraciones y/o locutores de interés, debe arrancarse el servidor de reconocimiento y/o síntesis pulsando el botón **Start**. En caso de disponer de licencias para todas las configuraciones y/o locutores seleccionados, *Verbio* se instalará como un servicio de Windows, arrancando conjuntamente con el sistema operativo si se ha seleccionado la opción `Automatic start-up` de la pestaña `Settings`. En caso contrario deberá arrancarse manualmente mediante `Verbio Server Configuration Manager` o mediante el panel de control de servicios de Windows.

Si no se dispone de licencias para alguna de las configuraciones y/o locutores seleccionados, *Verbio* arrancará en modo evaluación y finalizará transcurrido el intervalo de tiempo mostrado por pantalla. El semáforo de puesta en marcha continuará en rojo indicando que no se dispone de licencia y que, por lo tanto, no se ha arrancado en modo servicio. Sin embargo, es posible usar el sistema una vez aparezca por la pantalla de comandos la indicación `Started`.

6.7. Puesta en marcha del servidor de LOG

En caso de que se detecten problemas en el correcto funcionamiento de los sistemas de reconocimiento y síntesis del habla *Verbio*, es necesario recurrir al servidor de LOG para obtener trazas de las instrucciones recibidas, de modo que su análisis por parte de los técnicos de *VERBIO* pueda determinar la causa del mal funcionamiento del sistema. Para ello, debe seleccionarse el nivel de detalle que se aplicará a las trazas generadas (debe reiniciarse el servidor para que tenga en cuenta los cambios efectuados), que se almacenarán en la carpeta `$INSTALLDIR$\Verbio Engines\Log`, que deben ser enviados a la dirección de soporte técnico `support@verbio.com` para ser evaluados. En concreto, el fichero `verbiomsg.log` contendrá toda la información referente a transacciones procesadas por el servidor, mientras que el fichero `verbioexc.log` contendrá los errores críticos ocurridos en el servidor.

Si se desea modificar el nivel de detalle de las trazas generadas, es necesario detener el servicio, seleccionar el nuevo nivel de detalle y volver a iniciar el servicio. Es importante destacar que a mayor nivel de detalle, mayor es el coste computacional asociado al servicio de log, de modo que puede reducirse drásticamente el número de puertos soportados por una máquina.

En muchos casos es importante disponer de una visión del estado de instalación de *Verbio* en la máquina que presenta problemas. Como parte de los LOG, puede incluirse un resumen de la instalación seleccionando la opción `Start-up Trace` y reiniciando el servicio.

- `Connection level` no afecta a las prestaciones del sistema.
- `Function level` puede reducir en un 20% el número de canales soportados.
- `Detail level` puede reducir en un 50% el número de canales soportados

6.8. Detención y desinstalación del servicio

La detención del servidor de síntesis y/o reconocimiento del habla de *Verbio* puede hacerse desde `Verbio Server Configuration Manager` o desde el panel de control de servicios de Windows. En ambos casos debe pulsarse el botón **Stop**.

La desinstalación del servidor únicamente puede realizarse utilizando el desinstalador correspondiente desde `Agregar y quitar programas` del Panel de control de Windows. Es importante tener presente que la desinstalación de los distintos módulos debe realizarse en orden inverso a su instalación: en primer lugar

los asociados a configuraciones y locutores y, en último lugar, el asociado a los motores.

Atención

Antes de proceder a la actualización de versiones superiores, es imprescindible detener el servicio y desinstalar todos los componentes de la versión anterior siguiendo las recomendaciones del párrafo anterior.

Capítulo 3. Plataformas, herramientas y kits de desarrollo

1. Plataformas software y hardware

1.1. Integración de la telefonía y los ordenadores (CTI)

La amplia penetración de la telefonía en la sociedad (impulsada aún con más fuerza desde la aparición de la telefonía móvil) obliga a multitud de organizaciones a gestionar un gran volumen de llamadas (atención al cliente, campañas de márketing, etc.). La posibilidad de automatizar la atención de gran parte de estas llamadas llevó a la aparición de *hardware* específico: las tarjetas de telefonía, que incorporan interfaces para poder interactuar con cualquier protocolo telefónico, proporcionando un gran abanico de funcionalidades destinadas a procesar cualquier tipo de llamada. Esta nueva tecnología que explota la interacción entre los mundos de la informática y de la telefonía se conoce con el nombre de *Computer Telephony Integration (CTI)*.

Por otro lado, la constante evolución de los sistemas de reconocimiento del habla (ASR) y de síntesis del habla (TTS) ha permitido su incorporación en sistemas automáticos de recepción y emisión de llamadas telefónicas. Estos sistemas, que debido a su naturaleza eminentemente oral debieran haber sido el hábitat natural de las tecnologías del habla, recurrieron en sus inicios a estrategias tales como la reproducción de mensajes pregrabados y el reconocimiento por pulsación de tonos DTMF para interactuar con los usuarios. A pesar de que estas estrategias siguen siendo válidas e incluso aconsejables en algunos casos, actualmente las prestaciones obtenidas mediante la utilización conjunta o separada de sistemas ASR y/o TTS permiten dotar a los sistemas automatizados de una mayor naturalidad y fluidez en el proceso de intercambio de información con el usuario llamante.

La aparición en el mercado de una gran variedad de tarjetas telefónicas que se adaptan a cualquier necesidad y la evolución de los computadores actuales en cuanto a potencia de cálculo, han permitido que la tecnología CTI no requiera grandes inversiones y que, por lo tanto, también esté al alcance de las pequeñas y medianas empresas. Además, esta tecnología es fácilmente escalable, por lo que su implantación puede hacerse progresivamente en función de las necesidades de cada momento.

El desarrollo de *Verbio* ha tenido en cuenta que la mayor parte de las aplicaciones actuales de los sistemas de reconocimiento y síntesis del habla están desarrolladas para funcionar en entornos telefónicos. Es por ello que el sistema de reconocimiento incorporado en *Verbio* ha sido entrenado con señales de audio procedentes de entornos telefónicos, tanto fijos como móviles, con el objetivo de obtener las mejores tasas de reconocimiento posibles en este tipo de entornos. Adicionalmente, se dispone de modelos microfónicos (para alguno de los idiomas soportados) para dar salida al incremento de demanda de tecnologías del habla en el entorno de escritorio, domótico, industrial, etc.

Verbio ha sido utilizado con éxito por integradores de todo el mundo trabajando sobre tarjetas de los siguientes fabricantes (u otros, puesto que los SDK genéricos permiten interactuar con cualquier tarjeta):

1.1.1. Dialogic

Puede encontrar información de las tarjetas fabricadas por *Dialogic* en <http://www.dialogic.com>. *Verbio* permite ser incorporado en entornos *Dialogic* mediante el uso del entorno específico de programación *Dialogic SDK* o *CT-ADE SDK* o bien mediante los entornos genéricos *Library SDK* y *Advanced SDK*. En la Sección 2 se ofrece más información acerca de los entornos de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

1.1.2. Eicon (actualmente Dialogic)

Puede encontrar información de las tarjetas fabricadas por *Eicon Networks (Dialogic)* en <http://www.eicon.com>. *Verbio* permite ser incorporado en entornos *Eicon Diva Server* mediante el uso de los entornos de programación *Library SDK* o *Advanced SDK*. En la Sección 2 se ofrece más información acerca de los entornos de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

1.1.3. NMS

Puede encontrar información de las tarjetas fabricadas por Natural Microsystems en <http://www.nmss.com>. Verbio permite ser incorporado en entornos NMS mediante el uso de los entornos de programación *Library SDK* o *Advanced SDK*. En la Sección 2 se ofrece más información acerca de los entornos de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

1.1.4. Aculab

Puede encontrar información de las tarjetas fabricadas por Aculab en <http://www.aculab.com>. Verbio permite ser incorporado en entornos Aculab mediante el uso de los entornos de programación *Library SDK* o *Advanced SDK*. En la Sección 2 se ofrece más información acerca de los entornos de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

1.1.5. Avaya IR (Conversant)

Puede encontrar información de los sistemas comercializados por Avaya en <http://www.avaya.com>. Verbio permite ser incorporado en entornos Avaya IR (Conversant) mediante el uso del siguiente entorno de programación: *Avaya IR SDK*. En la Sección 2 se ofrece más información acerca de los distintos entornos de programación disponibles.

1.1.6. Altitude Software (Evox)

Puede encontrar información de los sistemas comercializados por Altitude Software en <http://www.altitude.com>. Verbio permite ser incorporado en entornos Altitude uCI mediante el uso del siguiente entorno de programación: *Evox SDK*. En la Sección 2 se ofrece más información acerca de los distintos entornos de programación disponibles.

1.1.7. Otras plataformas

Verbio permite ser utilizado conjuntamente con tarjetas CTI de otros fabricantes mediante los entornos de programación *Library SDK* o *Advanced SDK*. En ambos casos, el único requisito es que el desarrollador conozca suficientemente el API de programación de la tarjeta como para poder implementar toda la parte de adquisición y envío de muestras de/hacia la tarjeta CTI (intercambio de muestras con el sistema de reconocimiento de voz y síntesis de voz respectivamente).

Verbio ha desarrollado también un gateway *MRCP v1* y *MRCP v2*, de modo que cualquier plataforma del mercado que pueda comunicarse con un servidor *MRCP v1* o *MRCP v2* podrá utilizar los motores de reconocimiento y síntesis del habla de Verbio. Consulte con el departamento técnico de Verbio la disponibilidad de este producto, en support@verbio.com.

En la Sección 2 se ofrece más información acerca de los entornos de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

1.2. Programación de aplicaciones de escritorio

En el apartado anterior se hacía hincapié en que, actualmente, el mercado principal de los sistemas de reconocimiento y síntesis del habla es el mercado de las aplicaciones telefónicas. Sin embargo, estos sistemas pueden incorporarse en cualquier entorno en el que se disponga de dispositivos de adquisición de muestras de audio y de reproducción de muestras de audio. Típicamente, este dispositivo puede ser un ordenador personal con una tarjeta de sonido. Por lo tanto, también es posible desarrollar las aplicaciones de escritorio que interactúan con el usuario mediante un micrófono y unos altavoces.

Actualmente, Verbio ofrece modelos de reconocimiento globales específicamente adaptados a entornos microfónicos (escritorio, domótica) en algunos idiomas. No obstante, en caso de que se requieran modelos personalizados para otros idiomas, o bien para entornos muy específicos que requieran de unos modelos adaptados a los mismos, Verbio pone a disposición de sus clientes su capacidad para desarrollar modelos fonéticos a medida. De este modo, sin necesidad de cambios en las aplicaciones, el cliente dispone de modelos adaptados en cada momento a sus necesidades, con lo que se consiguen unas tasas de acierto muy elevadas. Por lo que respecta al conversor de texto en habla, éste sí dispone de una versión adaptada a las características de los entornos de escritorio en todos los idiomas actuales, lográndose una mayor calidad de síntesis respecto a los entornos telefónicos gracias al mayor ancho de banda disponible.

El desarrollo de aplicaciones de escritorio debe realizarse mediante la utilización de los entornos de programación *Library SDK* o *Advanced SDK*. En la Sección 2 se ofrece más información acerca de los entornos

de programación disponibles para poder seleccionar la opción que mejor se adapte en cada caso.

2. SDK - Kit de desarrollo de software

Verbio incluye un conjunto de librerías de desarrollo de aplicaciones (Software Development Kit - SDK) para facilitar la inclusión de recursos de reconocimiento y síntesis del habla en múltiples entornos. A continuación se describirán las particularidades de cada uno de los SDK existentes con la finalidad de que cada desarrollador encuentre aquel que mejor se adapte al entorno y hardware a utilizar.

2.1. Library SDK - Plataforma genérica

Library SDK proporciona un conjunto de funciones a bajo nivel que permiten utilizar cualquier dispositivo de audio (tarjetas CTI o tarjetas de sonido) siempre y cuando el desarrollador conozca cómo intercambiar muestras de audio con dicho dispositivo. Es decir, el desarrollador deberá encargarse de obtener las muestras de audio y pasarlas al sistema de reconocimiento (en procesos de reconocimiento del habla) o bien reproducir las muestras de audio que le entregue el sistema de síntesis (en procesos de síntesis del habla).

Las funciones de este SDK empiezan con el prefijo *vox_* y están disponibles desde entornos de programación en C/C++, aunque es posible su invocación directa a la DLL desde entornos distintos, como por ejemplo *Visual Basic*, *Delphi*, *Java*, etc. *Library SDK* se mantiene por compatibilidad con las antiguas versiones (a pesar de que se le han incorporado nuevas funcionalidades), aunque en caso de empezar a desarrollar una aplicación desde cero, es aconsejable utilizar la otra plataforma genérica *Advanced SDK*.

2.2. Advanced SDK - Plataforma genérica C++

Advanced SDK proporciona un conjunto de funciones a más alto nivel que las funciones contenidas en *Library SDK* sin que esto suponga perder el nivel de control en las funcionalidades que interese. De este modo, se pretende facilitar el uso de los sistemas de reconocimiento y síntesis del habla, especialmente en aquellos aspectos que suelen conllevar una mayor complejidad, tales como la gestión de los resultados devueltos por el reconocedor o la gestión de las desconexiones entre cliente y servidor. Existe una gran dualidad entre las funciones de ambos SDK, por lo que sería recomendable, siempre que sea posible, migrar de *Library SDK* a *Advanced SDK* para aprovechar las nuevas (y futuras) funcionalidades.

Advanced SDK se ha desarrollado a partir de *Library SDK*, incorporando elementos de *Dialogic SDK* con el fin de facilitar el uso de tarjetas *Dialogic* y añadiendo también funciones semejantes para facilitar el uso de tarjetas de otros fabricantes, tales como *Natural Microsystems* o *Eicon*.

Advanced SDK está formado por un conjunto de objetos definidos mediante clases C++, desarrollados inicialmente para ser utilizados desde *Microsoft Visual Studio*.

2.3. Dialogic SDK - Dialogic

Dialogic SDK proporciona un conjunto de funciones a bajo nivel especialmente diseñadas para facilitar la utilización de tarjetas *Dialogic*, debido a que internamente ya interactúan con la tarjeta CTI para obtener o enviar las muestras de audio según sea reconocimiento o síntesis del habla respectivamente.

Las funciones de este SDK empiezan con el prefijo *vx_* y están disponibles desde entornos de programación en C (o C++), aunque es posible su invocación directa a la DLL desde entornos distintos, como por ejemplo *Visual Basic*, *Delphi*, *Java*, etc. *Dialogic SDK* se mantiene por compatibilidad con las antiguas versiones (a pesar de que se le han incorporado nuevas funcionalidades), aunque en caso de empezar a desarrollar una aplicación desde cero, es aconsejable utilizar la otra plataforma genérica *Advanced SDK*, que dispone también de facilidades para interactuar con tarjetas *Dialogic*.

2.4. WordSpotting SDK

WordSpotting SDK proporciona un conjunto de funciones a bajo nivel especialmente diseñadas para desarrollar tareas de *WordSpotting* (búsqueda de palabras clave en contenidos de audio/video), debido a que internamente ya realiza la mayor parte de las operaciones necesarias para procesar los audios y entregar los resultados de forma sencilla de interpretar.

WordSpotting SDK está formado por un conjunto de objetos definidos mediante clases C++, desarrollados inicialmente para ser utilizados desde *Microsoft Visual Studio*. No obstante, existe una versión en funciones C estándar para ser utilizado desde otros entornos y lenguajes de programación. Estas funciones

empiezan con el prefijo `ws_`.

2.5. CT-ADE SDK

CT-ADE SDK proporciona un conjunto de funciones para ser utilizadas desde entornos de desarrollo Envoy CT ADE (antiguo VOS de Parity Software).

2.6. Avaya IR SDK

Avaya IR SDK proporciona un conjunto de funciones para ser utilizadas en entornos de desarrollo de Avaya IR (Conversant).

2.7. Evox SDK

Evox SDK proporciona un conjunto de funciones para ser utilizadas en entornos de desarrollo de Altitude Software uCI.

2.8. Microsoft SAPI 4/5 - TTS

Microsoft desarrolló el estándar *Speech API (SAPI)* con la finalidad de establecer patrones de uso de los motores de reconocimiento y síntesis desarrollados por cualquier fabricante. De este modo, cualquier aplicación puede hacer uso de cualquier motor de reconocimiento y/o síntesis compatible SAPI 4/5 sin ninguna dificultad. Todos los locutores incluidos en *Verbio* cumplen las especificaciones SAPI 4/5, de modo que están disponibles para ser utilizados mediante el SDK de Microsoft. Para más detalles, puede consultarse <http://www.microsoft.com/speech>

3. Herramientas del desarrollador

3.1. Verbio Grammar Manager

Verbio Grammar Manager: Herramienta destinada a diseñar y probar las gramáticas de reconocimiento utilizadas en la aplicación. Para más información consulte la documentación asociada *Verbio Software Reference: Guía del usuario de Grammar Manager*.

3.2. Verbio Read Aloud

Verbio Read Aloud: Herramienta destinada a probar los locutores disponibles mediante la especificación de cualquier texto y la manipulación de sus parámetros de funcionamiento básicos. Para más información consulte la documentación asociada *Verbio Software Reference: Guía del usuario de Read Aloud*.

Capítulo 4. Reconocimiento del habla

1. Introducción

El reconocimiento del habla permite que una aplicación tenga la funcionalidad de transformar la entrada de voz de un usuario en texto escrito. El reconocedor de *Verbio* es independiente de locutor, lo que significa que no se necesita entrenar o adaptar el sistema al usuario para que el reconocedor funcione. *Verbio* ha sido entrenado con una población de miles de hablantes de cada lengua en una proporción equilibrada en sexo y edad y de todas las partes del territorio de uso de cada lengua, recogiendo sus voces a través de la red telefónica fija y móvil para los modelos a 8KHz, así como micrófono para los modelos a 16 KHz.

Este capítulo explica cómo trabajar con el reconocedor de voz de *Verbio* y detalla los conceptos y pasos más importantes involucrados, que se resumen a continuación.

El reconocimiento del habla puede transformar la voz procedente del habla de un usuario a texto si conoce todas las posibles combinaciones de palabras que el usuario puede decir en un determinado momento. La descripción del vocabulario de palabras a usar por el reconocedor así como el conjunto de posibles secuencias de éstas en cualquier oración válida es lo que se conoce por *gramática* de reconocimiento (ver Sección 3).

Cuanto más se especializa una gramática, mejores pueden ser los resultados. En contrapartida, esto disminuye a menudo la flexibilidad y naturalidad en el estilo de habla de los usuarios. Por ejemplo, una gramática A para reconocimiento de fechas puede considerar fechas en formato estándar “21 de mayo”. Otra gramática B puede considerar además fechas relativas como “el viernes” o “el mes que viene, el cuatro”. La gramática B será obviamente más flexible, pero podría cometer más errores al intentar reconocer fechas en su formato estándar. Incluso una tercera gramática C podría considerar las horas y destinaciones para la reserva de vuelos además de las fechas. La flexibilidad en este caso será mucho mayor, pero las prestaciones del reconocimiento podrían disminuir frente a la situación mucho más controlada ofrecida por las gramáticas A y B. Las gramáticas de uso más habitual (fechas, números, respuestas de tipo sí/no, etc.), ya están definidas internamente en el sistema con tal de poder obtener los mejores resultados en todo momento.

Una vez se ha definido una gramática, existen varios parámetros de tiempo a considerar como: la máxima duración del silencio inicial, el silencio permitido una vez el usuario ya ha empezado a hablar y el tiempo máximo total (ver Sección 5).

Procesar el resultado no consiste solamente en obtener el texto resultante de la voz de entrada sino que es importante conocer una medida de confianza por parte del reconocedor (lo que permite rechazar resultados o, al menos, pedir confirmación al usuario), evaluando hipótesis alternativas (N-best) o procesando el resultado a través de una interpretación semántica (ver Sección 6).

2. Inicialización

La inicialización del motor de reconocimiento del habla consiste en habilitar uno o más idiomas para usar en la aplicación así como especificar el idioma por defecto para un vocabulario de reconocimiento (ver funciones de inicialización en la *Function Reference* para más detalles o la Sección 6.2 para especificar los valores por defecto).

3. Gramáticas de reconocimiento

Una gramática de reconocimiento define el universo de posibles respuestas que el sistema puede reconocer en un determinado momento. Es el subconjunto del “lenguaje” aceptado por el reconocedor. Por lo tanto debe haber una gramática adecuada para cada instante del diálogo entre la persona y la máquina. Una gramática especifica el vocabulario de palabras que el sistema puede reconocer así como las posibles secuencias en que éstas se pueden combinar.

3.1. Tipos de gramáticas

Por ejemplo, dado un vocabulario de palabras “cerrar, abrir, la, ventana, puerta”, una gramática podría ser “una de las palabras del vocabulario” (es decir, reconocimiento de palabras aisladas permitiendo que el usuario diga locuciones como “abrir” o “ventana” por ejemplo); otra gramática podría ser “una o varias de las palabras del vocabulario” (es decir, reconocimiento de palabras conectadas permitiendo al usuario decir locuciones como “puerta” o “abre la puerta”, pero también “la cerrar abrir”); finalmente, otra gramática “basada en patrones”

como “[abrir | cerrar] [la] [puerta | ventana]” imponiendo el lenguaje natural en una sintaxis de gramática de contexto libre, conocida como Modelo Estadístico del Lenguaje (SLM).

Los diferentes tipos de gramáticas definen el lenguaje y la terminología para expresar cómo combinar las palabras.

3.1.1. Listas de palabras

La gramática más sencilla para el reconocimiento del habla es la anteriormente mencionada como “una de las palabras del vocabulario”. *Verbio* permite usar ficheros de texto consistentes en listas de entradas (cada una con una o más palabras), una por línea. Cada una de estas entradas se denotará como *palabra del vocabulario*. El Ejemplo 4.1. Vocabulario de listas de palabras sencillo muestra un ejemplo de vocabulario de nombres de persona consistente en cuatro entradas o palabras del vocabulario.

Ejemplo 4.1. Vocabulario de listas de palabras sencillo

```
Luís Fernández
José Pérez
Pepe Pérez
María Sancho
```

Opcionalmente, las listas de palabras pueden formatearse en columnas separadas por un carácter tabular (“\t”). Entonces, la segunda columna de una entrada es la palabra en su forma literal, es decir, las palabras a pronunciar (transcribir fonéticamente). En este caso, la primera columna es el resultado devuelto cuando la palabra es reconocida. El Ejemplo 4.2. Vocabulario de listas de palabras avanzado es el mismo vocabulario de nombres de persona pero el resultado devuelto por el reconocedor es la extensión telefónica (nótese que dos entradas pueden devolver la misma palabra; “ -> ” indica tabulador (“\t”).

Las *palabras de relleno* (entradas que empiezan con el símbolo #) son útiles para indicar qué palabras acompañantes pueden usarse en el contexto de la gramática pero no son propiamente palabras con contenido (significantes) (una palabra de relleno reconocida no cuenta como palabra devuelta por las funciones de procesado del resultado).

Ejemplo 4.2. Vocabulario de listas de palabras avanzado

```
1023      ->  Luís Fernández
1024      ->  José Pérez
1024      ->  Pepe Pérez
1032      ->  María Sancho
#ponme con
#por favor
```

Estas gramáticas tipo “una de las palabras del vocabulario” se denotan en *Verbio* como *palabras aisladas* (ver flags ISOLATED en las Function Reference). Similarmente, las gramáticas tipo “una o varias de las palabras del vocabulario”, denotadas en *Verbio* como *palabras conectadas* (ver flags CONNECTED en las Function Reference) usan los mismos ficheros pero se permiten múltiples entradas por elocución del usuario.

3.1.1.1. Idioma

Las gramáticas *Verbio* soportan vocabularios plurilingües: algunas palabras pueden pronunciarse en un idioma y otras en otro distinto. Esto es particularmente interesante en directorios de nombres de persona y, especialmente, en territorios bilingües, donde los nombres acostumbran a pronunciarse acorde a su origen geográfico.

El idioma se especifica con los descriptores “[\$ESP]” (español), “[\$CAT]” (catalán), “[\$EUS]” (euskera), “[\$GAL]” (gallego), “[\$POR]” (portugués) y “[\$DEF]” (idioma por defecto). El idioma por defecto es el establecido al preparar el vocabulario.

En el Ejemplo 4.3. Vocabulario de listas de palabras bilingüe (idioma por defecto español), el idioma por defecto es español por lo que las palabras a pronunciar en catalán se delimitan con el descriptor [\$CAT], indicando las reglas de transcripción fonética a aplicar. Estos descriptores indican cómo procesar las palabras siguientes en la misma entrada del vocabulario y no afectan el procesado de las siguientes entradas (Nota: deben dejarse espacios en blanco antes y después de los descriptores de idioma).

Para más detalles sobre su implementación, ver documentación relacionada con el idioma en la inicialización y el establecimiento de parámetros en las `Function Reference`.

Ejemplo 4.3. Vocabulario de listas de palabras bilingüe (idioma por defecto español)

```
1023     ->  Lu s Fern ndez           ; en espa ol
1024     ->  Jos  P rez              ; en espa ol
1024     ->  Pepe P rez              ; en espa ol
1027     ->  [$CAT] Andreu Pujol      ; en catal n
1027     ->  Andr s [$CAT] Pujol      ; "Andr s" en espa ol, "Pujol" en catal n
1029     ->  [$CAT] Jordi [$DEF] Sevilla ; "Jordi" en catal n, "Sevilla" en espa ol
1032     ->  Mar a Sancho            ; en espa ol
#ponme con
#por favor
```

3.1.2. Gram ticas ABNF

Como se mencionaba anteriormente, las gram ticas de reconocimiento pueden verse como una estructura basada en patrones que describen las posibles secuencias de palabras del vocabulario v lidas. Estas gram ticas especifican c mo se comporta el *lenguaje natural* en una determinada tarea.

Las gram ticas de *Verbio* siguen la sintaxis descrita en la especificaci n de gram ticas para el reconocimiento del habla (SRGS, del ingl s *Speech Recognition Grammar Specification*) por el W3C *Speech Interface Framework* (<http://www.w3c.org/TR/speech-grammar>) en su forma *Augmented BNF (ABNF)*.

Recomendamos usar la herramienta *Verbio Grammar Manager* para editar, compilar y probar las gram ticas *Verbio* ABNF.

3.1.2.1. Formato de gram ticas ABNF

Este apartado resume la SRGS (*Speech Recognition Grammar Specification*) en su forma ABNF. Consulte la especificaci n para m s detalles.

Una gram tica define el patr n de palabras a reconocer, una regla consistente en una expresi n regular que contiene palabras, referencias a otras reglas o combinaciones de ambas.

3.1.2.1.1. Tokens

Un token es la parte de la gram tica que define las palabras que pueden pronunciarse, las anteriormente definidas palabras del vocabulario. En la sintaxis, un token es cualquier combinaci n de caracteres delimitados por espacios o s mbolos especiales (`;` `=` `|` `()` `{}` `*` `+` `/*` `*/` `//` `<>` `!` `"`). Si un token contiene espacios o s mbolos especiales debe entrecomillarse (`"``"`).

El s mbolo `"_"` entre dos palabras indica que en la transcripci n fon tica se tendr n especialmente en cuenta los fen menos de coarticulaci n entre palabras. Esta opci n no deber a usarse a menos que exista una particularmente alta coarticulaci n entre palabras (por ejemplo la coarticulaci n de `"y"` en n meros como 34, `"treinta_y_cuatro"`).

```
Hola           // token
ponme con Juan // 3 tokens diferentes
"buenos d as" // 1 token
treinta_y_dos  // 1 token
"[SIL]"        // 1 token: [SIL]
```

3.1.2.1.2. Reglas

Cada regla se identifica con un nombre de regla. Una referencia a una regla local (definida en la misma gram tica) se define con su nombre de regla, que debe empezar siempre con el car cter `"$"`.

```
$dia
$mes
```

3.1.2.1.3. Secuencias

Una secuencia de tokens, reglas o combinaciones de ambas es a su vez una regla, lo que implica el orden cronol gico en el que las palabras pueden pronunciarse.

```
ponme con Juan // secuencia de 3 tokens
ponme con $nombre // secuencia de 2 tokens y 1 referencia a una regla
$relleno $nombre // secuencia de referencias a reglas
```

3.1.2.1.4. Alternativas

Un conjunto de alternativas también es una regla. Se identifican como una lista de reglas separadas por el símbolo “ | ”.

```
Juan | José | "José Luis" | $nombres
```

Cada alternativa puede opcionalmente tener un peso (valor en coma flotante entre “ / ”).

```
/2/ Juan | /3/ José | /1/ "José Luis"
```

Este peso es proporcional a la probabilidad de inserción de dicha alternativa en el reconocimiento de una elocución. Por defecto, el peso es 1.0

Un peso mayor que 1 aumenta la probabilidad de inserción de un token comparado con el resto de palabras. Un peso menor que 1 implica una mayor probabilidad de eliminar dicha palabra del reconocimiento.

Si todas las palabras tienen un peso superior a 1, la probabilidad de inserción de palabras es más alta. Por lo tanto, una lista de alternativas con todos los pesos a 1 no es equivalente a la misma lista con todos los pesos a 10, por ejemplo. En este último caso, la probabilidad de inserción de palabras se vería incrementada.

Si el peso se asigna a una regla o expresión, dicho peso multiplicará a los pesos de todas las palabras contenidas en dicha regla. Por lo tanto, la regla:

```
/2/ Juan | /3/ José | /3/ (/1/ Luis | /2/ Jorge)
```

es equivalente a:

```
/2/ Juan | /3/ José | /3/ Luis | /6/ Jorge
```

Por lo tanto, es conveniente usar los pesos con cuidado y preferiblemente en las palabras finales exclusivamente.

3.1.2.1.5. Reglas especiales

GARBAGE Define una regla que puede mapear cualquier segmento de habla hasta que lo haga la siguiente regla o bien el final de la elocución.

SILENCE Define una regla que puede mapear un silencio. Es útil para indicar silencios particularmente largos dentro de una elocución. Por defecto, los silencios al principio y final de la elocución ya se insertan automáticamente en el procesador de gramáticas. Entre palabras, y también de forma automática, se insertan silencios cortos opcionales.

```
// "ponme con Juan"
// "por favor podrías ponerme con José"
// "Luis"
$GARBAGE (Juan | José | Luis)
```

3.1.2.1.6. Repeticiones: opcional, *, +

Palabras o reglas opcionales se delimitan por corchetes [. . .]. Los símbolos * y + indican que la palabra o regla precedente puede repetirse cero o más veces o una o más veces respectivamente.

```
// "1"
// "1 4 3 2"
$digit +

// "pizza"
// "pizza grande con queso"
// "pizza pequeña con queso y anchoas"
pizza [[muy] grande | pequeña] ([con | y] $complemento)*
```

3.1.2.1.7. Tags

Cada token puede asociarse a una cadena de texto que el reconocedor devuelve en lugar de la palabra pronunciada. El tag se delimita entre llaves { . . . }.

```
["ponme con" {}] ("Luis Fernández" {1023} | "José Pérez" {1024} | "Pepe Pérez" {1024})
```

Cuando el tag está vacío “ {} ”, el token se considera como una palabra de relleno y no será jamás devuelta por el reconocedor. En el ejemplo anterior, el resultado del reconocedor ante la elocución "ponme con José Pérez" sería "1024".

3.1.2.1.8. Idioma

Las gramáticas ABNF soportan vocabularios plurilingües, es decir, palabras de la gramática que se pronunciarán en un idioma distinto al establecido por defecto. Siempre hay un idioma por defecto con el que el vocabulario se prepara (para más detalles sobre su implementación, ver documentación relacionada con el idioma en la inicialización y el establecimiento de parámetros en las `Function Reference`).

El idioma por defecto de un vocabulario o gramática puede establecerse `inline` usando el descriptor `language` seguido por el identificador de idioma (`es` para español, `ca` para catalán, `eu` para euskera, `ga` para gallego, `es-ar` para español argentino, `es-cl` para español chileno, `es-co` para español colombiano, `es-mx` para español mexicano, `es-ve` para español venezolano, `pt` para portugués, `pt-br` para brasileño, `fr` para francés y `en-us` para inglés americano) y un punto y coma.

El idioma de los tokens puede establecerse añadiendo después del token el símbolo de admiración `!` seguido del identificador de idioma.

```
language es;
$color =
    verde | azul | rojo |           // en español
    verd!ca | blau!ca | vermell!ca // en catalán
;
```

3.1.2.1.9. Precedencia

La precedencia de operadores en la definición de reglas es:

1. Nombre de regla denotado por el símbolo `$`.
2. Paréntesis `()` para agrupar y corchetes `[]` para agrupar opcionales.
3. Operadores unarios `*` y `+` y tags `{ }` aplican a la palabra o regla inmediatamente precedente (los paréntesis permiten agrupar palabras y reglas).
4. Secuencias de palabras y/o reglas.
5. Secuencias de palabras y/o reglas separadas por `|`.

3.1.2.1.10. Definición de reglas

Una definición de regla permite asociar una regla a un nombre de regla. Consiste en una declaración opcional de ámbito seguido de un nombre de regla (que debe empezar obligatoriamente por `$`), un signo igual `=`, la propia definición de la regla, y un punto y coma `;` al final.

```
$nombre = regla;
public $nombre = regla;
$color = rojo | verde | azul | "azul marino";
```

La declaración del ámbito puede ser `public` o `private`, siendo esta última la declaración por defecto. Las reglas definidas como `public` son las reglas iniciales para el reconocimiento.

3.1.2.1.11. Regla principal (root)

Si una (y sólo una) de las reglas se declara con el ámbito `root`, será la única regla usada por el reconocedor. Si no existe ninguna regla declarada como `root`, el reconocedor usará como regla principal la alternativa de todas las reglas declaradas como `public`.

3.1.2.1.12. Comentarios

Comentarios en el estilo `C/C++/Java` se admiten en la definición de las gramáticas. Los comentarios delimitados por `/*` y `*/` o entre `//` y el final de la línea son ignorados por el procesador de gramáticas.

El Ejemplo 4.4. Ejemplos de gramáticas ABNF ilustra la sintaxis de las gramáticas ABNF. Notése que todas deben empezar por `#ABNF 1.0 ISO8859-1;`.

Ejemplo 4.4. Ejemplos de gramáticas ABNF

```
#ABNF 1.0 ISO8859-1;
```

```

$rgb = rojo {COLOR=R} | // si se reconoce "rojo" retorna COLOR=R
         verde {COLOR=G} | // si se reconoce "verde" retorna COLOR=G
         azul {COLOR=B} |
         "azul marino" {COLOR=B};

// "el color azul" -> COLOR=B
// "el rojo por favor" -> COLOR=R
// "azul marino" -> COLOR=B
root $color = [color {} | "el color" {} | el {}] $rgb "por favor" {};

#ABNF 1.0 ISO8859-1;

public $dia_semana = [el {}] (lunes | martes | miércoles | jueves | viernes | sábado | domingo);
public $dia_relativo = ayer | hoy | mañana | "pasado mañana";

// implicit declaration of rule "root $dia = $dia_semana | $dia_rel;"

```

3.1.3. Gramáticas básicas incorporadas (built-in)

Existen gramáticas especialmente diseñadas para las tareas más comunes (y a menudo difíciles) que han sido integradas en el reconocedor como un recurso incorporado. Las gramáticas básicas incorporadas no son solamente la definición de las reglas sino el procesado interno de los resultados considerando medidas de confianza y opcionalmente múltiples hipótesis (N-Best).

Hay que tener en cuenta que estas gramáticas pueden incorporar modelos de ruido y de palabras de relleno, de modo que están pensadas para aplicaciones reales en las que el usuario puede decir alguna palabra no contemplada en la gramática sin perjudicar la locución reconocida. Además, de este modo son más robustas al ruido captado por el teléfono. Este hecho puede hacer perder un poco de prestaciones en entornos muy silenciosos o de laboratorio si se comparan con sistemas que no contemplan los casos anteriores.

En función de la gramática y del entorno de trabajo, será necesario elegir la versión de la `built-in` con o sin modelo de relleno (GARBAGE). El nombre de la gramática `built-in` indica si la gramática incluye el modelo de GARBAGE o no (en este caso, el nombre finaliza con el sufijo `ng`). Por ejemplo, la gramática `boolean` incluye modelo de relleno. Sin embargo, la gramática `booleanng` no lo incluye.

Las gramáticas `built-in` deben utilizarse con el modelo de relleno siempre que se utilicen de forma aislada (sin otras gramáticas cargadas simultáneamente) y en entornos con niveles de ruido habituales o elevados.

Las gramáticas `built-in` deben utilizarse SIN el modelo de relleno en aquellos casos en los que formen parte de una gramática ABNF que también deba reconocer otros ítems, en caso de utilizarse en paralelo con otras gramáticas o en caso de condiciones de ruido mínimo o inexistente, donde el usuario pronuncia, únicamente, alguna de las alternativas previstas por la gramática.

<code>boolean</code> - <code>booleanng</code>	respuestas a preguntas tipo sí/no, cubriendo las palabras “sí”, “no” (con modelado acústico específico), y las palabras y expresiones más comunes para expresar confirmación y negación (“exacto”, “correcto”, “no gracias”, etc.).
<code>digits</code> - <code>digitsng</code>	cadena de dígitos de longitud limitada o no, usando modelos acústicos específicos para los dígitos conectados (“uno, tres, cuatro”, “siete”, etc.).
<code>date</code> - <code>dateng</code>	fechas en lenguaje natural, cubriendo fechas fijas y relativas (“dentro de tres días”, “el lunes”, etc.).
<code>number</code> - <code>numberng</code>	números naturales dentro de un rango limitado (“dos mil trescientos cuarenta y siete”, “doce”, etc.).
<code>phone</code> - <code>phoneng</code>	números de teléfono en lenguaje natural (para numeración española), cubriendo las más frecuentes formas de agrupar los dígitos en números naturales (“noventa y tres, cuatro cero nueve, setenta y uno, veinte”).
<code>time</code> - <code>timeng</code>	horas y minutos, así como horas relativas (“a las tres y veinticinco”, etc.).
<code>currency</code> - <code>currencyng</code>	cantidades monetarias (“veintisiete euros con cinco céntimos”)
<code>creditcard</code> - <code>creditcardng</code>	números de tarjetas de crédito (dictados dígito a dígito)
<code>code</code> - <code>codeng</code>	códigos numéricos hablados como una secuencia de números naturales
<code>nif</code> - <code>nifng</code>	número de identificación fiscal personal (NIF) de España

spell - spellng deletreos, tanto alfabéticos como numéricos

Tabla 4.1. Built-in soportadas en cada configuración

	es	ca	eu	ga	pt	pt-br	fr	en-us	es-mx	es-ar	es-co	es-cl	es-ve
boolean	X	X	X	X	X	X	X	X	X	X	X	X	X
code	X	X	X		X	X	X	X	X	X	X	X	X
creditcard	X	X	X	X	X	X	X	X	X	X	X	X	X
currency	X	X	X		X	X	X	X	X	X	X	X	X
date	X	X	X		X	X	X	X	X	X	X	X	X
digits	X	X	X	X	X	X	X	X	X	X	X	X	X
nif	X	X	X	X									
number	X	X	X		X	X	X	X	X	X	X	X	X
phone	X	X	X		*	*		*		X			
spell	X	X	X	X				X					
time	X	X	X						X	X	X	X	X
* números telefónicos del territorio español													

Estas gramáticas tienen a menudo parámetros para establecer opciones como la mínima y/o la máxima longitud de una cadena de dígitos, el rango de valores de un número, o la flexibilidad de una fecha. Estas opciones permiten la posibilidad de ajustar la gramática más conveniente en base al contexto o a la pregunta formulada y obtener así los mejores resultados.

Las gramáticas básicas incorporadas se denotan de la siguiente forma:

```
builtin:grammar/name:language?param1=value1;param2=value2;...;paramN=valueN
```

Por ejemplo:

```
builtin:grammar/number:es?max=2000 (para entornos que requieran modelo de relleno)
builtin:grammar/numberng:es?max=2000 (para entornos que NO requieran modelo de relleno)
```

Las gramáticas `builtin` pueden usarse desde los distintos SDK como si fueran una gramática ABNF convencional, con la única salvedad de que *no hay que preparar el vocabulario*, puesto que ya viene preparado. Únicamente hay que establecer o activar la gramática para que esté operativa. Por ejemplo:

```
vx_loadvcb(chdev, "builtin:grammar/date:es?relative=true;year=false", GVX_ABNF); //Dialogic SDK
vox_setvcb(dev, "builtin:grammar/booleanng:ca", GVX_ABNF); //Library SDK
EngineASRRResource->LoadGrammar("builtin:grammar/nif:es", VERBIO_GRAMMAR_ABNF); //Advanced SDK
```

Asimismo, también pueden incluirse las gramáticas `builtin` dentro de gramáticas ABNF convencionales. Por ejemplo:

```
#ABNF 1.0 ISO8859-1;
root $item = $help | $spell;
$spell = $(builtin:grammar/spellng:es?minlength=1;maxlength=1;alphanumeric=false);
$help = ayuda {HELP} | repetir {REPEAT};
```

Nota

Atención, cuando se incluyen gramáticas `builtin` dentro de gramáticas ABNF, debe especificarse siempre el idioma en el que se usará la `builtin`, siguiendo la sintaxi especificada anteriormente.

Los siguientes apartados detallan los parámetros para cada una de las gramáticas básicas incorporadas, así como la interpretación semántica del resultado.

3.1.3.1. boolean

Respuestas a preguntas tipo sí/no, cubriendo las palabras “sí”, “no” (con modelado acústico específico), y las palabras y expresiones más comunes para expresar confirmación y negación (“exacto”, “correcto”, “no gracias”, etc.).

Esta gramática no tiene parámetros.

Como resultado del reconocimiento, la gramática boolean devuelve “true” (respuesta afirmativa) o “false” (respuesta negativa).

3.1.3.2. digits

Cadenas de dígitos de longitud limitada o no, usando modelos acústicos específicos para los dígitos conectados.

Los parámetros para esta gramática son:

minlength	Valores: int Defecto: 1 Definición: La longitud mínima de la cadena de dígitos.
maxlength	Valores: int Defecto: 10 Definición: La longitud máxima de la cadena de dígitos.
length	Valores: int Defecto: - Definición: La longitud exacta de la cadena de dígitos (minlength=maxlength).

La gramática digits devuelve la cadena de dígitos (de “0” a “9”) reconocida sin espacios. Por ejemplo: “1972”.

3.1.3.3. date

Fechas en lenguaje natural, cubriendo fechas fijas y/o relativas (“dentro de dos días”, “el próximo lunes”) en función de los parámetros establecidos.

Los parámetros para esta gramática son:

relative	Valores: true, false Defecto: false Definición: Si el valor es “true”, la gramática acepta fechas relativas como “mañana” y/o incompletas como “el sábado”.
year	Valores: true, false Defecto: false Definición: La gramática acepta fechas fijas con día y mes. Si el valor de este parámetro es “true”, la gramática acepta además opcionalmente un año.
voicexml	Valores: true, false Defecto: false Definición: Si el parámetro es “true”, el formato de salida es yyyyymmdd, usando "?" en caso de que no se hayan rellenado todos los campos.

Establecer a “true” estos dos parámetros aumenta la flexibilidad en el modo de decir las fechas, y no empeora en ninguno de los casos las tasas de reconocimiento de elocuciones con solamente día y mes en formato estándar. Las diferencias se reducen al coste computacional. Las gramáticas aumentan el coste computacional a medida que aumenta su rango de cobertura (y por lo tanto su complejidad), tal y como se detalla en la Sección 4.1.

La gramática `date` devuelve la fecha en el formato “`swddmmyyy`” rellenando únicamente aquellos valores que se hayan reconocido. Los caracteres que no se hayan podido rellenar tendrán el valor “?”.

<code>s</code>	Carácter especial: “+”, “-” para fechas relativas o “E” para fechas especiales
<code>w</code>	Día de la semana: de “1” para el lunes hasta “7” para el domingo. Es “0” en el caso de “la semana que viene” (+0???????) o en el caso de “la semana pasada” (-0???????)
<code>dd</code>	Día del mes: de “01” a “31”
<code>mm</code>	Mes: de “01” a “12”
<code>yyyy</code>	Año: a partir de “1900”

Ejemplos: “?12305????” (“lunes 23 de mayo”), “+??02????” (“dentro de 2 meses”), “E?0601????” (“el día de reyes”).

3.1.3.4. number

Números naturales dentro de un rango limitado.

Los parámetros para esta gramática son:

<code>max</code>	Valores: int
	Defecto: 99
	Definición: Valor numérico máximo a devolver. La gramática no puede devolver valores superiores a 999999.

La gramática `number` devuelve el número reconocido, sin separación entre millares y centenas. Por ejemplo: “23710”.

3.1.3.5. phone

Números de teléfono en lenguaje natural, cubriendo las más frecuentes formas de agrupar los dígitos en números naturales (“noventa y tres, cuatro cero nueve, setenta y uno, veinte”).

Los parámetros para esta gramática son:

<code>minlength</code>	Valores: int
	Defecto: 9
	Definición: El mínimo número de dígitos del número de teléfono.
<code>maxlength</code>	Valores: int
	Defecto: 9
	Definición: El máximo número de dígitos del número de teléfono.

La gramática `phone` devuelve el número de teléfono reconocido como una cadena de dígitos (de “0” a “9”). Por ejemplo: “934097120”.

3.1.3.6. time

Horas (horas y minutos) en lenguaje natural , así como horas relativas (“a las tres y veintinco”, etc.)

La gramática `time` devuelve la fecha en el formato “hhmmx” rellenando únicamente aquellos valores que se hayan reconocido. Los caracteres que no se hayan podido rellenar tendrán el valor “?”.

hh Horas: de "00" a "23"

mm Minutos: de "00" a "59"

x franja horaria: "a" si se ha especificado explícitamente que es horario de mañana, "p" si se ha especificado explícitamente que es horario de tarde, "h" si se ha dicho una hora de tarde (de "13" a "24") o "?" en cualquier otro caso.

Ejemplos: "0930?" ("nueve y media"), "2245p" ("once menos cuarto de la noche"), "17:35h" ("diecisiete treinta y cinco").

3.1.3.7. number

Números naturales dentro de un rango limitado.

Los parámetros para esta gramática son:

max Valores: int

Defecto: 99

Definición: Valor numérico máximo a devolver. La gramática no puede devolver valores superiores a 999999.

La gramática `number` devuelve el número reconocido, sin separación entre millares y centenas. Por ejemplo: “23710”.

3.1.3.8. currency

Cantidades monetarias expresadas en Euros. En caso de que se incluyan céntimos, éstos ocuparán dos decimales separados por un punto de las unidades de Euro. El formato de salida es, por ejemplo, “EUR125.39”.

Esta gramática no tiene parámetros.

3.1.3.9. creditcard

Devuelve el número de la tarjeta de crédito reconocido (dictado cifra a cifra) con todas las cifras numéricas sin espacios intermedios.

Esta gramática no tiene parámetros.

3.1.3.10. code

códigos numéricos dictados como una secuencia de números naturales (agrupados en grupos de una, dos o tres cifras). Por ejemplo: "veintitrés, cinco, diez" devuelve "23510".

Los parámetros para esta gramática son:

length Valores: int

Defecto: 4

Definición: Longitud máxima del código numérico a devolver.

La gramática `code` devuelve el código numérico reconocido con todas las cifras juntas, sin espacios intermedios.

3.1.3.11. nif

Devuelve el Número de Identificación Fiscal español, incluida la letra, sin espacios intermedios.

Esta gramática no tiene parámetros.

3.1.3.12. spell

Cadenas alfabéticas o alfanuméricas sin espacios intermedios.

Los parámetros para esta gramática son:

<code>minlength</code>	Valores: <code>int</code> Defecto: <code>1</code> Definición: La longitud mínima de la cadena alfa/numérica.
<code>maxlength</code>	Valores: <code>int</code> Defecto: <code>4</code> Definición: La longitud máxima de la cadena alfa/numérica.
<code>spontaneous</code>	Valores: <code>true, false</code> Defecto: <code>false</code> Definición: Indicado cuando el deletreo es realizado de forma muy rápida (no aconsejable).
<code>alphanumeric</code>	Valores: <code>true, false</code> Defecto: <code>true</code> Definición: Sólo caracteres alfabéticos (<code>false</code>) o bien alfanuméricos (<code>true</code>).

La gramática `spell` devuelve la cadena alfa/numérica reconocida sin espacios intermedios.

3.1.4. Modelos estadísticos del Lenguaje (SLM) - Verbio VoxPopuli

A partir de la versión 8 de 8, pueden utilizarse modelos estadísticos del lenguaje (SLM) para acometer tareas donde el habla espontánea sea necesaria o requerida. La naturaleza independiente del locutor de los motores de reconocimiento de *Verbio* condiciona el ámbito de trabajo de esta tecnología; es decir, no será posible el reconocimiento abierto y espontáneo de cualquier locución de cualquier temática, sino que previamente habrá que definir una temática de trabajo y desarrollar un modelo estadístico que la cubra de la forma más completa posible.

El mecanismo de creación de un modelo estadístico consiste en obtener una gran cantidad de texto que haga referencia a la temática que se desea modelar para, posteriormente, poder ser capaz de transcribir conversaciones o locuciones que traten de la misma. A título de ejemplo, un caso que suele dar muy buenos resultados es el dictado de informes médicos de una especialidad concreta: se parte de miles de informes previos digitalizados de la especialidad elegida y, a partir de ellos, se crea un modelo estadístico. Su uso desde *Verbio VoxPopuli* permitirá al especialista dictar el informe en lugar de escribirlo, con la consiguiente ganancia en rapidez y automatización.

Una vez *Verbio* dispone de los textos, evaluará si hay suficiente material de entreno y si la tarea es suficientemente acotada como para que el modelo resultante pueda dar una buena tasa de acierto (a nivel de palabra) cuando se utilice para la transcripción automática. En caso positivo, *Verbio* creará el modelo estadístico y lo entregará al cliente con instrucciones de uso de cómo utilizarlo en sus aplicaciones.

Para facilitar al máximo el uso de los modelos estadísticos, el funcionamiento de *Verbio VoxPopuli* es muy similar, en cuanto a programación, al funcionamiento de *Verbio ASR* con gramáticas convencionales. De hecho, se usan las mismas funciones de los SDK para establecer gramáticas convencionales y modelos estadísticos, así como para obtener los resultados de reconocimiento. Para más detalles acerca de cómo usar los modelos estadísticos, consulte las guías de programación o contacte con support@verbio.com.

3.1.5. Otras gramáticas para el reconocimiento del habla

La especificación de gramáticas para el reconocimiento del habla (SRGS) en su forma XML pueden usarse si se traducen a su forma ABNF usando XSLT tal y como se detalla en <http://www.w3c.org/TR/speech-grammar>. *Verbio* lo hace automáticamente en los casos más sencillos, puesto que internamente utiliza el formato ABNF. Es por ello que, en nuevos desarrollos, y siempre que sea posible, se aconseja utilizar el formato ABNF.

3.1.6. Modelos específicos de palabra

El reconocedor de habla *Verbio* contiene modelos acústicos específicos para las palabras más frecuentes y críticas en funcionamiento como son los dígitos (de 0 a 9) aislados y conectados, las palabras sí/no y las letras para el deletreo.

Las gramáticas básicas incorporadas o built-in usan estos modelos de palabras con tal de incrementar las prestaciones en el reconocimiento de cadenas de dígitos y respuestas tipo sí/no. Por ejemplo, una posible gramática para el reconocimiento de números NIF podría ser:

```
#ABNF 1.0 ISO8859-1;

root $nif = $DIGITOS $LETRA;

// gramáticas auxiliares

$DIGITOS = ( ( $S | $SD $DD* $DS ) "[SIL]" {} )+ ;

$LETRA =

    "[A_es][_A_es]" {A} | // a
    "[B_es][_E_es]" {B} | // be
    "[C_es][_E_es]" {C} | // ce
    "[D_es][_E_es]" {D} | // de
    "[E_es][_E_es]" {E} | // e
    "[E_es][F_es][_e_es]" {F} | // efe
    "[G_es][_E_es]" {G} | // ge
    "[H_es]" {H} | // hache
    "[I_es]" {I} | // i
    "[Il_es]" {I} | // i latina
    "[J_es]" {J} | // jota
    "[K_es][_A_es]" {K} | // ca
    "[E_es][L_es][_e_es]" {L} | // ele
    "[E_es][M_es][_e_es]" {M} | // eme
    "[E_es][N_es][_e_es]" {N} | // ene
    "[O_es]" {O} | // o
    "[P_es][_E_es]" {P} | // pe
    "[Q_es][_U_es]" {Q} | // cu
    "[E_es][R_es][_e_es]" {R} | // erre
    "[E_es][r_es][_e_es]" {R} | // ere
    "[E_es][S_es][_e_es]" {S} | // ese
    "[T_es][_E_es]" {T} | // te
    "[U_es][_U_es]" {U} | // u
    "[V_es]" {V} | // uve
    "[W_es]" {W} | // uve doble
    "[dV_es]" {W} | // doble uve
    "[X_es]" {X} | // equis
    "[Y_es]" {Y} | // y griega
    "[Z_es]" {Z}; // zeta

// variables auxiliares

$S =

    "[S-cero_es][cero_es+S]" {0} |
    "[S-uno_es][uno_es+S]" {1} |
    "[S-dos_es][dos_es+S]" {2} |
    "[S-tres_es][tres_es+S]" {3} |
    "[S-cuatro_es][cuatro_es+S]" {4} |
    "[S-cinco_es][cinco_es+S]" {5} |
    "[S-seis_es][seis_es+S]" {6} |
    "[S-siete_es][siete_es+S]" {7} |
    "[S-ocho_es][ocho_es+S]" {8} |
    "[S-nueve_es][nueve_es+S]" {9} ;

$SD =

    "[S-cero_es][cero_es+D]" {0} |
    "[S-uno_es][uno_es+D]" {1} |
    "[S-dos_es][dos_es+D]" {2} |
    "[S-tres_es][tres_es+D]" {3} |
    "[S-cuatro_es][cuatro_es+D]" {4} |
    "[S-cinco_es][cinco_es+D]" {5} |
    "[S-seis_es][seis_es+D]" {6} |
    "[S-siete_es][siete_es+D]" {7} |
    "[S-ocho_es][ocho_es+D]" {8} |
    "[S-nueve_es][nueve_es+D]" {9} ;
```



```

$DS =
    "[D-cero_es][cero_es+S]"      {0}
    "[D-uno_es][uno_es+S]"        {1}
    "[D-dos_es][dos_es+S]"        {2}
    "[D-tres_es][tres_es+S]"       {3}
    "[D-cuatro_es][cuatro_es+S]"  {4}
    "[D-cinco_es][cinco_es+S]"    {5}
    "[D-seis_es][seis_es+S]"       {6}
    "[D-siete_es][siete_es+S]"     {7}
    "[D-ocho_es][ocho_es+S]"       {8}
    "[D-nueve_es][nueve_es+S]"    {9} ;

$DD =
    "[D-cero_es][cero_es+D]"       {0}
    "[D-uno_es][uno_es+D]"         {1}
    "[D-dos_es][dos_es+D]"         {2}
    "[D-tres_es][tres_es+D]"       {3}
    "[D-cuatro_es][cuatro_es+D]"   {4}
    "[D-cinco_es][cinco_es+D]"     {5}
    "[D-seis_es][seis_es+D]"       {6}
    "[D-siete_es][siete_es+D]"     {7}
    "[D-ocho_es][ocho_es+D]"       {8}
    "[D-nueve_es][nueve_es+D]"    {9} ;

```

Tabla 4.2. Modelos específicos de palabras para español castellano

<i>Palabras</i>	
[si_es]	Modela la palabra "sí"
[no_es]	Modela la palabra "no"
<i>Deletreos</i>	
[A_es][_A_es]	Modela la letra A
[B_es][_E_es]	Modela la letra B
[C_es][_E_es]	Modela la letra C
[Ch_es][_E_es]	Modela la letra CH (che)
[CH_es]	Modela la letra CH (ce hache)
[D_es][_E_es]	Modela la letra D
[E_es][_E_es]	Modela la letra E
[E__es][F_es][_e_es]	Modela la letra F
[G_es][_E_es]	Modela la letra G
[H_es]	Modela la letra H
[I_es]	Modela la letra I
[Il_es]	Modela la locución "I latina"
[J_es]	Modela la letra J
[K_es][_A_es]	Modela la letra K
[E__es][L_es][_e_es]	Modela la letra L
[E_es][Ll_es][_e_es]	Modela la letra LL (elle)
[dL_es]	Modela la letra LL (doble ele)
[Ld_es]	Modela la letra LL (ele doble)
[LL_es]	Modela la letra LL (ele ele)
[E__es][M_es][_e_es]	Modela la letra M
[E__es][N_es][_e_es]	Modela la letra N
[E__es][NY_es][_e_es]	Modela la letra Ñ
[O_es]	Modela la letra O
[P_es][_E_es]	Modela la letra P
[Q_es][_U_es]	Modela la letra Q
[E__es][R_es][_e_es]	Modela la letra R (erre)
[E__es][r_es][_e_es]	Modela la letra R (ere)
[E__es][S_es][_e_es]	Modela la letra S

[T_es][_E_es]	Modela la letra T
[U_es][_U_es]	Modela la letra U
[V_es]	Modela la letra V
[W_es]	Modela la letra W (uve doble)
[dV_es]	Modela la letra W (doble uve)
[X_es]	Modela la letra X
[Y_es]	Modela la letra Y
[Z_es]	Modela la letra Z
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es][cero_es+S]	Dígito 0
[S-uno_es][uno_es+S]	Dígito 1
[S-dos_es][dos_es+S]	Dígito 2
[S-tres_es][tres_es+S]	Dígito 3
[S-cuatro_es][cuatro_es+S]	Dígito 4
[S-cinco_es][cinco_es+S]	Dígito 5
[S-seis_es][seis_es+S]	Dígito 6
[S-siete_es][siete_es+S]	Dígito 7
[S-ocho_es][ocho_es+S]	Dígito 8
[S-nueve_es][nueve_es+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es][cero_es+D]	Dígito 0
[S-uno_es][uno_es+D]	Dígito 1
[S-dos_es][dos_es+D]	Dígito 2
[S-tres_es][tres_es+D]	Dígito 3
[S-cuatro_es][cuatro_es+D]	Dígito 4
[S-cinco_es][cinco_es+D]	Dígito 5
[S-seis_es][seis_es+D]	Dígito 6
[S-siete_es][siete_es+D]	Dígito 7
[S-ocho_es][ocho_es+D]	Dígito 8
[S-nueve_es][nueve_es+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es][cero_es+S]	Dígito 0
[D-uno_es][uno_es+S]	Dígito 1
[D-dos_es][dos_es+S]	Dígito 2
[D-tres_es][tres_es+S]	Dígito 3
[D-cuatro_es][cuatro_es+S]	Dígito 4
[D-cinco_es][cinco_es+S]	Dígito 5
[D-seis_es][seis_es+S]	Dígito 6
[D-siete_es][siete_es+S]	Dígito 7
[D-ocho_es][ocho_es+S]	Dígito 8
[D-nueve_es][nueve_es+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es][cero_es+D]	Dígito 0
[D-uno_es][uno_es+D]	Dígito 1
[D-dos_es][dos_es+D]	Dígito 2
[D-tres_es][tres_es+D]	Dígito 3

[D-cuatro_es][cuatro_es+D]	Dígito 4
[D-cinco_es][cinco_es+D]	Dígito 5
[D-seis_es][seis_es+D]	Dígito 6
[D-siete_es][siete_es+D]	Dígito 7
[D-ocho_es][ocho_es+D]	Dígito 8
[D-nueve_es][nueve_es+D]	Dígito 9

Tabla 4.3. Modelos específicos de palabras para catalán

<i>Palabras</i>	
[si_ca]	Modela la palabra "sí"
[no_ca]	Modela la palabra "no"
<i>Deletreos</i>	
[A_ca]	Modela la letra A
[B_ca]	Modela la letra B
[B_ca][alta_ca]	Modela la letra B (be alta)
[C_ca]	Modela la letra C
[C_ca][trencada_ca]	Modela la letra Ç
[D_ca]	Modela la letra D
[E_ca]	Modela la letra E
[F_ca]	Modela la letra F
[G_ca]	Modela la letra G
[H_ca]	Modela la letra H
[I_ca]	Modela la letra I
[J_ca]	Modela la letra J
[K_ca]	Modela la letra K
[L_ca]	Modela la letra L
[LL_ca]	Modela la letra LL (ella)
[doble_ca][L_ca]	Modela la letra LL (doble ela)
[L_ca][doble_ca]	Modela la letra LL (ela doble)
[M_ca]	Modela la letra M
[N_ca]	Modela la letra N
[NY_ca]	Modela la letra NY
[O_ca]	Modela la letra O
[P_ca]	Modela la letra P
[Q_ca]	Modela la letra Q
[R_ca]	Modela la letra R (erre)
[S_ca]	Modela la letra S
[T_ca]	Modela la letra T
[U_ca]	Modela la letra U
[V_ca]	Modela la letra V
[B_ca][baixa_ca]	Modela la letra V (be baixa)
[uve_ca]	Modela la letra V (uve)
[B_ca][doble_ca]	Modela la letra W (be doble)
[doble_ca][B_ca]	Modela la letra W (doble be)

[X_ca]	Modela la letra X
[I_ca][grega_ca]	Modela la letra Y
[Z_ca]	Modela la letra Z
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_ca][zero_ca+S]	Dígito 0
[S-u_ca][u_ca+S]	Dígito 1
[S-dos_es][dos_es+S]	Dígito 2
[S-tres_ca][tres_ca+S]	Dígito 3
[S-quatre_ca][quatre_ca+S]	Dígito 4
[S-cinc_ca][cinc_ca+S]	Dígito 5
[S-sis_ca][sis_ca+S]	Dígito 6
[S-set_ca][set_ca+S]	Dígito 7
[S-vuit_es][vuit_es+S]	Dígito 8
[S-nou_ca][nou_ca+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_ca][zero_ca+D]	Dígito 0
[S-u_ca][u_ca+D]	Dígito 1
[S-dos_es][dos_es+D]	Dígito 2
[S-tres_ca][tres_ca+D]	Dígito 3
[S-quatre_ca][quatre_ca+D]	Dígito 4
[S-cinc_ca][cinc_ca+D]	Dígito 5
[S-sis_ca][sis_ca+D]	Dígito 6
[S-set_ca][set_ca+D]	Dígito 7
[S-vuit_es][vuit_es+D]	Dígito 8
[S-nou_ca][nou_ca+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_ca][zero_ca+S]	Dígito 0
[D-u_ca][u_ca+S]	Dígito 1
[D-dos_es][dos_es+S]	Dígito 2
[D-tres_ca][tres_ca+S]	Dígito 3
[D-quatre_ca][quatre_ca+S]	Dígito 4
[D-cinc_ca][cinc_ca+S]	Dígito 5
[D-sis_ca][sis_ca+S]	Dígito 6
[D-set_ca][set_ca+S]	Dígito 7
[D-vuit_es][vuit_es+S]	Dígito 8
[D-nou_ca][nou_ca+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-zero_ca][zero_ca+D]	Dígito 0
[D-u_ca][u_ca+D]	Dígito 1
[D-dos_es][dos_es+D]	Dígito 2
[D-tres_ca][tres_ca+D]	Dígito 3
[D-quatre_ca][quatre_ca+D]	Dígito 4
[D-cinc_ca][cinc_ca+D]	Dígito 5
[D-sis_ca][sis_ca+D]	Dígito 6
[D-set_ca][set_ca+D]	Dígito 7
[D-vuit_es][vuit_es+D]	Dígito 8

[D-nou_ca][nou_ca+D]	Dígito 9
----------------------	----------

Tabla 4.4. Modelos específicos de palabras para euskera

<i>Palabras</i>	
[bai_eu]	Modela la palabra "bai"
[ez_eu]	Modela la palabra "ez"
<i>Deletreos</i>	
[A_eu]	Modela la letra A
[B_eu]	Modela la letra B
[C1_eu]	Modela la letra C (X e)
[C2_eu]	Modela la letra C (T e)
[D_eu]	Modela la letra D
[E_eu]	Modela la letra E
[F_eu]	Modela la letra F
[G2_eu]	Modela la letra G
[H_eu]	Modela la letra H
[I_eu]	Modela la letra I
[J1_eu]	Modela la letra J (dj o t a)
[J2_eu]	Modela la letra J (x o t a)
[K_eu]	Modela la letra K
[L_eu]	Modela la letra L
[M_eu]	Modela la letra M
[N_eu]	Modela la letra N
[NY_eu]	Modela la letra Ñ
[O_eu]	Modela la letra O
[P_eu]	Modela la letra P
[Q_eu]	Modela la letra Q
[R_eu]	Modela la letra R (erre)
[S_eu]	Modela la letra S
[T_eu]	Modela la letra T
[U_eu]	Modela la letra U
[V_eu]	Modela la letra V
[W1_eu]	Modela la letra W (u B e B i k o i tX a)
[W2_eu]	Modela la letra W (u B e d o B l e)
[W3_eu]	Modela la letra W (d o B l e u B e)
[X1_eu]	Modela la letra X (i S a)
[X2_eu]	Modela la letra X (e k i s)
[X3_eu]	Modela la letra X (i k s a)
[Y1_eu]	Modela la letra Y (i G r r e k o a)
[Y2_eu]	Modela la letra Y (i G r r i e G a)
[Y3_eu]	Modela la letra Y (i G r r e k o t a r r a)
[Y4_eu]	Modela la letra Y (i G r r e X i a r r a)
[Y5_eu]	Modela la letra Y (i G r r e k e r a)
[Z1_eu]	Modela la letra Z (X e t a)

[Z2_eu]	Modela la letra Z (s e t a)
[Z3_eu]	Modela la letra Z (T e t a)
[Z4_eu]	Modela la letra Z (X e D a)
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_eu][zero_eu+S] [S-huts_eu][huts_eu+S] [S-hutsa_eu][hutsa_eu+S]	Dígito 0
[S-bat_eu][bat_eu+S]	Dígito 1
[S-bi_eu][bi_eu+S]	Dígito 2
[S-hiru_eu][hiru_eu+S]	Dígito 3
[S-lau_eu][lau_eu+S]	Dígito 4
[S-bost_eu][bost_eu+S]	Dígito 5
[S-sei_eu][sei_eu+S]	Dígito 6
[S-zazpi_eu][zazpi_eu+S]	Dígito 7
[S-zortzi_eu][zortzi_eu+S]	Dígito 8
[S-bederatzi_eu][bederatzi_eu+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_eu][zero_eu+D]	Dígito 0
[S-bat_eu][bat_eu+D]	Dígito 1
[S-bi_eu][bi_eu+D]	Dígito 2
[S-hiru_eu][hiru_eu+D]	Dígito 3
[S-lau_eu][lau_eu+D]	Dígito 4
[S-bost_eu][bost_eu+D]	Dígito 5
[S-sei_eu][sei_eu+D]	Dígito 6
[S-zazpi_eu][zazpi_eu+D]	Dígito 7
[S-zortzi_eu][zortzi_eu+D]	Dígito 8
[S-bederatzi_eu][bederatzi_eu+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_eu][zero_eu+S]	Dígito 0
[D-bat_eu][bat_eu+S]	Dígito 1
[D-bi_eu][bi_eu+S]	Dígito 2
[D-hiru_eu][hiru_eu+S]	Dígito 3
[D-lau_eu][lau_eu+S]	Dígito 4
[D-bost_eu][bost_eu+S]	Dígito 5
[D-sei_eu][sei_eu+S]	Dígito 6
[D-zazpi_eu][zazpi_eu+S]	Dígito 7
[D-zortzi_eu][zortzi_eu+S]	Dígito 8
[D-bederatzi_eu][bederatzi_eu+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-zero_eu][zero_eu+D]	Dígito 0
[D-bat_eu][bat_eu+D]	Dígito 1
[D-bi_eu][bi_eu+D]	Dígito 2
[D-hiru_eu][hiru_eu+D]	Dígito 3
[D-lau_eu][lau_eu+D]	Dígito 4
[D-bost_eu][bost_eu+D]	Dígito 5

[D-sei_eu][sei_eu+D]	Dígito 6
[D-zazpi_eu][zazpi_eu+D]	Dígito 7
[D-zortzi_eu][zortzi_eu+D]	Dígito 8
[D-bederatzi_eu][bederatzi_eu+D]	Dígito 9

Tabla 4.5. Modelos específicos de palabras para gallego

<i>Palabras</i>	
[si_ga]	Modela la palabra "sí"
[non_ga]	Modela la palabra "non"
<i>Deletreos</i>	
[A_ga]	Modela la letra A
[B_ga]	Modela la letra B
[C_ga]	Modela la letra C
[CH_ga]	Modela la letra CH
[D_ga]	Modela la letra D
[E_ga]	Modela la letra E
[F_ga]	Modela la letra F
[G_ga]	Modela la letra G
[H_ga]	Modela la letra H
[I_ga]	Modela la letra I
[J_ga]	Modela la letra J
[K_ga]	Modela la letra K
[L_ga]	Modela la letra L
[LL_ga]	Modela la letra LL
[dobre_ga][L_ga]	Modela la letra LL
[L_ga][dobre_ga]	Modela la letra LL
[M_ga]	Modela la letra M
[N_ga]	Modela la letra N
[NY_ga]	Modela la letra NY
[O_ga]	Modela la letra O
[P_ga]	Modela la letra P
[Q_ga]	Modela la letra Q
[R_ga]	Modela la letra R (erre)
[S_ga]	Modela la letra S
[T_ga]	Modela la letra T
[U_ga]	Modela la letra U
[V_ga]	Modela la letra V
[V_ga][dobre_ga]	Modela la letra W
[dobre_ga][V_ga]	Modela la letra W
[X_ga]	Modela la letra X
[Y_ga]	Modela la letra Y
[Z_ga]	Modela la letra Z
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_ga][cero_ga+S]	Dígito 0

[S-un_ga][un_ga+S]	Dígito 1
[S-dous_ga][dous_ga+S]	Dígito 2
[S-tres_ga][tres_ga+S]	Dígito 3
[S-catro_ga][catro_ga+S]	Dígito 4
[S-cinco_ga][cinco_ga+S]	Dígito 5
[S-seis_ga][seis_ga+S]	Dígito 6
[S-sete_ga][sete_ga+S]	Dígito 7
[S-oito_ga][oito_ga+S]	Dígito 8
[S-nove_ga][nove_ga+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_ga][cero_ga+D]	Dígito 0
[S-un_ga][un_ga+D]	Dígito 1
[S-dous_ga][dous_ga+D]	Dígito 2
[S-tres_ga][tres_ga+D]	Dígito 3
[S-catro_ga][catro_ga+D]	Dígito 4
[S-cinco_ga][cinco_ga+D]	Dígito 5
[S-seis_ga][seis_ga+D]	Dígito 6
[S-sete_ga][sete_ga+D]	Dígito 7
[S-oito_ga][oito_ga+D]	Dígito 8
[S-nove_ga][nove_ga+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_ga][cero_ga+S]	Dígito 0
[D-un_ga][un_ga+S]	Dígito 1
[D-dous_ga][dous_ga+S]	Dígito 2
[D-tres_ga][tres_ga+S]	Dígito 3
[D-catro_ga][catro_ga+S]	Dígito 4
[D-cinco_ga][cinco_ga+S]	Dígito 5
[D-seis_ga][seis_ga+S]	Dígito 6
[D-sete_ga][sete_ga+S]	Dígito 7
[D-oito_ga][oito_ga+S]	Dígito 8
[D-nove_ga][nove_ga+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_ga][cero_ga+D]	Dígito 0
[D-un_ga][un_ga+D]	Dígito 1
[D-dous_ga][dous_ga+D]	Dígito 2
[D-tres_ga][tres_ga+D]	Dígito 3
[D-catro_ga][catro_ga+D]	Dígito 4
[D-cinco_ga][cinco_ga+D]	Dígito 5
[D-seis_ga][seis_ga+D]	Dígito 6
[D-sete_ga][sete_ga+D]	Dígito 7
[D-oito_ga][oito_ga+D]	Dígito 8
[D-nove_ga][nove_ga+D]	Dígito 9

Tabla 4.6. Modelos específicos de palabras para portugués

<i>Palabras</i>	
[sim_pt]	Modela la palabra "sim"
[nao_pt]	Modela la palabra "não"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_pt][zero_pt+S]	Dígito 0
[S-um_pt][um_pt+S]	Dígito 1
[S-uma_pt][uma_pt+S]	Dígito 1
[S-dois_pt][dois_pt+S]	Dígito 2
[S-tres_pt][tres_pt+S]	Dígito 3
[S-quatro_pt][quatro_pt+S]	Dígito 4
[S-cinco_pt][cinco_pt+S]	Dígito 5
[S-seis_pt][seis_pt+S]	Dígito 6
[S-sete_pt][sete_pt+S]	Dígito 7
[S-oito_pt][oito_pt+S]	Dígito 8
[S-nove_pt][nove_pt+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_pt][zero_pt+D]	Dígito 0
[S-um_pt][um_pt+D]	Dígito 1
[S-dois_pt][dois_pt+D]	Dígito 2
[S-tres_pt][tres_pt+D]	Dígito 3
[S-quatro_pt][quatro_pt+D]	Dígito 4
[S-cinco_pt][cinco_pt+D]	Dígito 5
[S-seis_pt][seis_pt+D]	Dígito 6
[S-sete_pt][sete_pt+D]	Dígito 7
[S-oito_pt][oito_pt+D]	Dígito 8
[S-nove_pt][nove_pt+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_pt][zero_pt+S]	Dígito 0
[D-um_pt][um_pt+S]	Dígito 1
[D-dois_pt][dois_pt+S]	Dígito 2
[D-tres_pt][tres_pt+S]	Dígito 3
[D-quatro_pt][quatro_pt+S]	Dígito 4
[D-cinco_pt][cinco_pt+S]	Dígito 5
[D-seis_pt][seis_pt+S]	Dígito 6
[D-sete_pt][sete_pt+S]	Dígito 7
[D-oito_pt][oito_pt+S]	Dígito 8
[D-nove_pt][nove_pt+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-zero_pt][zero_pt+D]	Dígito 0
[D-um_pt][um_pt+D]	Dígito 1
[D-dois_pt][dois_pt+D]	Dígito 2
[D-tres_pt][tres_pt+D]	Dígito 3
[D-quatro_pt][quatro_pt+D]	Dígito 4
[D-cinco_pt][cinco_pt+D]	Dígito 5
[D-seis_pt][seis_pt+D]	Dígito 6
[D-sete_pt][sete_pt+D]	Dígito 7

[D-oito_pt][oito_pt+D]	Dígito 8
[D-nove_pt][nove_pt+D]	Dígito 9

Tabla 4.7. Modelos específicos de palabras para portugués brasileño

<i>Palabras</i>	
[sim_pt-br]	Modela la palabra "sim"
[nao_pt-br]	Modela la palabra "não"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_pt-br][zero_pt-br+S]	Dígito 0
[S-um_pt-br][um_pt-br+S]	Dígito 1
[S-uma_pt-br][uma_pt-br+S]	Dígito 1
[S-dois_pt-br][dois_pt-br+S]	Dígito 2
[S-tres_pt-br][tres_pt-br+S]	Dígito 3
[S-quatro_pt-br][quatro_pt-br+S]	Dígito 4
[S-cinco_pt-br][cinco_pt-br+S]	Dígito 5
[S-seis_pt-br][seis_pt-br+S]	Dígito 6
[S-meia_pt-br][meia_pt-br+S]	Dígito 6
[S-sete_pt-br][sete_pt-br+S]	Dígito 7
[S-oito_pt-br][oito_pt-br+S]	Dígito 8
[S-nove_pt-br][nove_pt-br+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_pt-br][zero_pt-br+D]	Dígito 0
[S-um_pt-br][um_pt-br+D]	Dígito 1
[S-uma_pt-br][uma_pt-br+D]	Dígito 1
[S-dois_pt-br][dois_pt-br+D]	Dígito 2
[S-tres_pt-br][tres_pt-br+D]	Dígito 3
[S-quatro_pt-br][quatro_pt-br+D]	Dígito 4
[S-cinco_pt-br][cinco_pt-br+D]	Dígito 5
[S-seis_pt-br][seis_pt-br+D]	Dígito 6
[S-meia_pt-br][meia_pt-br+D]	Dígito 6
[S-sete_pt-br][sete_pt-br+D]	Dígito 7
[S-oito_pt-br][oito_pt-br+D]	Dígito 8
[S-nove_pt-br][nove_pt-br+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_pt-br][zero_pt-br+S]	Dígito 0
[D-um_pt-br][um_pt-br+S]	Dígito 1
[D-uma_pt-br][uma_pt-br+S]	Dígito 1
[D-dois_pt-br][dois_pt-br+S]	Dígito 2
[D-tres_pt-br][tres_pt-br+S]	Dígito 3
[D-quatro_pt-br][quatro_pt-br+S]	Dígito 4
[D-cinco_pt-br][cinco_pt-br+S]	Dígito 5
[D-seis_pt-br][seis_pt-br+S]	Dígito 6
[D-meia_pt-br][meia_pt-br+S]	Dígito 6
[D-sete_pt-br][sete_pt-br+S]	Dígito 7

[D-oito_pt-br][oito_pt-br+S]	Dígito 8
[D-nove_pt-br][nove_pt-br+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-zero_pt-br][zero_pt-br+D]	Dígito 0
[D-um_pt-br][um_pt-br+D]	Dígito 1
[D-uma_pt-br][uma_pt-br+D]	Dígito 1
[D-dois_pt-br][dois_pt-br+D]	Dígito 2
[D-tres_pt-br][tres_pt-br+D]	Dígito 3
[D-quatro_pt-br][quatro_pt-br+D]	Dígito 4
[D-cinco_pt-br][cinco_pt-br+D]	Dígito 5
[D-seis_pt-br][seis_pt-br+D]	Dígito 6
[D-meia_pt-br][meia_pt-br+D]	Dígito 6
[D-sete_pt-br][sete_pt-br+D]	Dígito 7
[D-oito_pt-br][oito_pt-br+D]	Dígito 8
[D-nove_pt-br][nove_pt-br+D]	Dígito 9

Tabla 4.8. Modelos específicos de palabras para español argentino

<i>Palabras</i>	
[si_es-ar]	Modela la palabra "sí"
[no_es-ar]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es-ar][cero_es-ar+S]	Dígito 0
[S-uno_es-ar][uno_es-ar+S]	Dígito 1
[S-dos_es-ar][dos_es-ar+S]	Dígito 2
[S-tres_es-ar][tres_es-ar+S]	Dígito 3
[S-cuatro_es-ar][cuatro_es-ar+S]	Dígito 4
[S-cinco_es-ar][cinco_es-ar+S]	Dígito 5
[S-seis_es-ar][seis_es-ar+S]	Dígito 6
[S-siete_es-ar][siete_es-ar+S]	Dígito 7
[S-ocho_es-ar][ocho_es-ar+S]	Dígito 8
[S-nueve_es-ar][nueve_es-ar+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es-ar][cero_es-ar+D]	Dígito 0
[S-uno_es-ar][uno_es-ar+D]	Dígito 1
[S-dos_es-ar][dos_es-ar+D]	Dígito 2
[S-tres_es-ar][tres_es-ar+D]	Dígito 3
[S-cuatro_es-ar][cuatro_es-ar+D]	Dígito 4
[S-cinco_es-ar][cinco_es-ar+D]	Dígito 5
[S-seis_es-ar][seis_es-ar+D]	Dígito 6
[S-siete_es-ar][siete_es-ar+D]	Dígito 7
[S-ocho_es-ar][ocho_es-ar+D]	Dígito 8
[S-nueve_es-ar][nueve_es-ar+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es-ar][cero_es-ar+S]	Dígito 0

[D-uno_es-ar][uno_es-ar+S]	Dígito 1
[D-dos_es-ar][dos_es-ar+S]	Dígito 2
[D-tres_es-ar][tres_es-ar+S]	Dígito 3
[D-cuatro_es-ar][cuatro_es-ar+S]	Dígito 4
[D-cinco_es-ar][cinco_es-ar+S]	Dígito 5
[D-seis_es-ar][seis_es-ar+S]	Dígito 6
[D-siete_es-ar][siete_es-ar+S]	Dígito 7
[D-ocho_es-ar][ocho_es-ar+S]	Dígito 8
[D-nueve_es-ar][nueve_es-ar+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es-ar][cero_es-ar+D]	Dígito 0
[D-uno_es-ar][uno_es-ar+D]	Dígito 1
[D-dos_es-ar][dos_es-ar+D]	Dígito 2
[D-tres_es-ar][tres_es-ar+D]	Dígito 3
[D-cuatro_es-ar][cuatro_es-ar+D]	Dígito 4
[D-cinco_es-ar][cinco_es-ar+D]	Dígito 5
[D-seis_es-ar][seis_es-ar+D]	Dígito 6
[D-siete_es-ar][siete_es-ar+D]	Dígito 7
[D-ocho_es-ar][ocho_es-ar+D]	Dígito 8
[D-nueve_es-ar][nueve_es-ar+D]	Dígito 9

Tabla 4.9. Modelos específicos de palabras para español chileno

<i>Palabras</i>	
[si_es-cl]	Modela la palabra "sí"
[no_es-cl]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es-cl][cero_es-cl+S]	Dígito 0
[S-uno_es-cl][uno_es-cl+S]	Dígito 1
[S-dos_es-cl][dos_es-cl+S]	Dígito 2
[S-tres_es-cl][tres_es-cl+S]	Dígito 3
[S-cuatro_es-cl][cuatro_es-cl+S]	Dígito 4
[S-cinco_es-cl][cinco_es-cl+S]	Dígito 5
[S-seis_es-cl][seis_es-cl+S]	Dígito 6
[S-siete_es-cl][siete_es-cl+S]	Dígito 7
[S-ocho_es-cl][ocho_es-cl+S]	Dígito 8
[S-nueve_es-cl][nueve_es-cl+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es-cl][cero_es-cl+D]	Dígito 0
[S-uno_es-cl][uno_es-cl+D]	Dígito 1
[S-dos_es-cl][dos_es-cl+D]	Dígito 2
[S-tres_es-cl][tres_es-cl+D]	Dígito 3
[S-cuatro_es-cl][cuatro_es-cl+D]	Dígito 4
[S-cinco_es-cl][cinco_es-cl+D]	Dígito 5
[S-seis_es-cl][seis_es-cl+D]	Dígito 6

[S-siete_es-cl][siete_es-cl+D]	Dígito 7
[S-ocho_es-cl][ocho_es-cl+D]	Dígito 8
[S-nueve_es-cl][nueve_es-cl+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es-cl][cero_es-cl+S]	Dígito 0
[D-uno_es-cl][uno_es-cl+S]	Dígito 1
[D-dos_es-cl][dos_es-cl+S]	Dígito 2
[D-tres_es-cl][tres_es-cl+S]	Dígito 3
[D-cuatro_es-cl][cuatro_es-cl+S]	Dígito 4
[D-cinco_es-cl][cinco_es-cl+S]	Dígito 5
[D-seis_es-cl][seis_es-cl+S]	Dígito 6
[D-siete_es-cl][siete_es-cl+S]	Dígito 7
[D-ocho_es-cl][ocho_es-cl+S]	Dígito 8
[D-nueve_es-cl][nueve_es-cl+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es-cl][cero_es-cl+D]	Dígito 0
[D-uno_es-cl][uno_es-cl+D]	Dígito 1
[D-dos_es-cl][dos_es-cl+D]	Dígito 2
[D-tres_es-cl][tres_es-cl+D]	Dígito 3
[D-cuatro_es-cl][cuatro_es-cl+D]	Dígito 4
[D-cinco_es-cl][cinco_es-cl+D]	Dígito 5
[D-seis_es-cl][seis_es-cl+D]	Dígito 6
[D-siete_es-cl][siete_es-cl+D]	Dígito 7
[D-ocho_es-cl][ocho_es-cl+D]	Dígito 8
[D-nueve_es-cl][nueve_es-cl+D]	Dígito 9

Tabla 4.10. Modelos específicos de palabras para español colombiano

<i>Palabras</i>	
[si_es-co]	Modela la palabra "sí"
[no_es-co]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es-co][cero_es-co+S]	Dígito 0
[S-uno_es-co][uno_es-co+S]	Dígito 1
[S-dos_es-co][dos_es-co+S]	Dígito 2
[S-tres_es-co][tres_es-co+S]	Dígito 3
[S-cuatro_es-co][cuatro_es-co+S]	Dígito 4
[S-cinco_es-co][cinco_es-co+S]	Dígito 5
[S-seis_es-co][seis_es-co+S]	Dígito 6
[S-siete_es-co][siete_es-co+S]	Dígito 7
[S-ocho_es-co][ocho_es-co+S]	Dígito 8
[S-nueve_es-co][nueve_es-co+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es-co][cero_es-co+D]	Dígito 0
[S-uno_es-co][uno_es-co+D]	Dígito 1

[S-dos_es-co][dos_es-co+D]	Dígito 2
[S-tres_es-co][tres_es-co+D]	Dígito 3
[S-cuatro_es-co][cuatro_es-co+D]	Dígito 4
[S-cinco_es-co][cinco_es-co+D]	Dígito 5
[S-seis_es-co][seis_es-co+D]	Dígito 6
[S-siete_es-co][siete_es-co+D]	Dígito 7
[S-ocho_es-co][ocho_es-co+D]	Dígito 8
[S-nueve_es-co][nueve_es-co+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es-co][cero_es-co+S]	Dígito 0
[D-uno_es-co][uno_es-co+S]	Dígito 1
[D-dos_es-co][dos_es-co+S]	Dígito 2
[D-tres_es-co][tres_es-co+S]	Dígito 3
[D-cuatro_es-co][cuatro_es-co+S]	Dígito 4
[D-cinco_es-co][cinco_es-co+S]	Dígito 5
[D-seis_es-co][seis_es-co+S]	Dígito 6
[D-siete_es-co][siete_es-co+S]	Dígito 7
[D-ocho_es-co][ocho_es-co+S]	Dígito 8
[D-nueve_es-co][nueve_es-co+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es-co][cero_es-co+D]	Dígito 0
[D-uno_es-co][uno_es-co+D]	Dígito 1
[D-dos_es-co][dos_es-co+D]	Dígito 2
[D-tres_es-co][tres_es-co+D]	Dígito 3
[D-cuatro_es-co][cuatro_es-co+D]	Dígito 4
[D-cinco_es-co][cinco_es-co+D]	Dígito 5
[D-seis_es-co][seis_es-co+D]	Dígito 6
[D-siete_es-co][siete_es-co+D]	Dígito 7
[D-ocho_es-co][ocho_es-co+D]	Dígito 8
[D-nueve_es-co][nueve_es-co+D]	Dígito 9

Tabla 4.11. Modelos específicos de palabras para español mexicano

<i>Palabras</i>	
[si_es-mx]	Modela la palabra "sí"
[no_es-mx]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es-mx][cero_es-mx+S]	Dígito 0
[S-uno_es-mx][uno_es-mx+S]	Dígito 1
[S-dos_es-mx][dos_es-mx+S]	Dígito 2
[S-tres_es-mx][tres_es-mx+S]	Dígito 3
[S-cuatro_es-mx][cuatro_es-mx+S]	Dígito 4
[S-cinco_es-mx][cinco_es-mx+S]	Dígito 5
[S-seis_es-mx][seis_es-mx+S]	Dígito 6
[S-siete_es-mx][siete_es-mx+S]	Dígito 7

[S-ocho_es-mx][ocho_es-mx+S]	Dígito 8
[S-nueve_es-mx][nueve_es-mx+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es-mx][cero_es-mx+D]	Dígito 0
[S-uno_es-mx][uno_es-mx+D]	Dígito 1
[S-dos_es-mx][dos_es-mx+D]	Dígito 2
[S-tres_es-mx][tres_es-mx+D]	Dígito 3
[S-cuatro_es-mx][cuatro_es-mx+D]	Dígito 4
[S-cinco_es-mx][cinco_es-mx+D]	Dígito 5
[S-seis_es-mx][seis_es-mx+D]	Dígito 6
[S-siete_es-mx][siete_es-mx+D]	Dígito 7
[S-ocho_es-mx][ocho_es-mx+D]	Dígito 8
[S-nueve_es-mx][nueve_es-mx+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es-mx][cero_es-mx+S]	Dígito 0
[D-uno_es-mx][uno_es-mx+S]	Dígito 1
[D-dos_es-mx][dos_es-mx+S]	Dígito 2
[D-tres_es-mx][tres_es-mx+S]	Dígito 3
[D-cuatro_es-mx][cuatro_es-mx+S]	Dígito 4
[D-cinco_es-mx][cinco_es-mx+S]	Dígito 5
[D-seis_es-mx][seis_es-mx+S]	Dígito 6
[D-siete_es-mx][siete_es-mx+S]	Dígito 7
[D-ocho_es-mx][ocho_es-mx+S]	Dígito 8
[D-nueve_es-mx][nueve_es-mx+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es-mx][cero_es-mx+D]	Dígito 0
[D-uno_es-mx][uno_es-mx+D]	Dígito 1
[D-dos_es-mx][dos_es-mx+D]	Dígito 2
[D-tres_es-mx][tres_es-mx+D]	Dígito 3
[D-cuatro_es-mx][cuatro_es-mx+D]	Dígito 4
[D-cinco_es-mx][cinco_es-mx+D]	Dígito 5
[D-seis_es-mx][seis_es-mx+D]	Dígito 6
[D-siete_es-mx][siete_es-mx+D]	Dígito 7
[D-ocho_es-mx][ocho_es-mx+D]	Dígito 8
[D-nueve_es-mx][nueve_es-mx+D]	Dígito 9

Tabla 4.12. Modelos específicos de palabras para español venezolano

<i>Palabras</i>	
[si_es-ve]	Modela la palabra "sí"
[no_es-ve]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-cero_es-ve][cero_es-ve+S]	Dígito 0
[S-uno_es-ve][uno_es-ve+S]	Dígito 1
[S-dos_es-ve][dos_es-ve+S]	Dígito 2

[S-tres_es-ve][tres_es-ve+S]	Dígito 3
[S-cuatro_es-ve][cuatro_es-ve+S]	Dígito 4
[S-cinco_es-ve][cinco_es-ve+S]	Dígito 5
[S-seis_es-ve][seis_es-ve+S]	Dígito 6
[S-siete_es-ve][siete_es-ve+S]	Dígito 7
[S-ocho_es-ve][ocho_es-ve+S]	Dígito 8
[S-nueve_es-ve][nueve_es-ve+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-cero_es-ve][cero_es-ve+D]	Dígito 0
[S-uno_es-ve][uno_es-ve+D]	Dígito 1
[S-dos_es-ve][dos_es-ve+D]	Dígito 2
[S-tres_es-ve][tres_es-ve+D]	Dígito 3
[S-cuatro_es-ve][cuatro_es-ve+D]	Dígito 4
[S-cinco_es-ve][cinco_es-ve+D]	Dígito 5
[S-seis_es-ve][seis_es-ve+D]	Dígito 6
[S-siete_es-ve][siete_es-ve+D]	Dígito 7
[S-ocho_es-ve][ocho_es-ve+D]	Dígito 8
[S-nueve_es-ve][nueve_es-ve+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-cero_es-ve][cero_es-ve+S]	Dígito 0
[D-uno_es-ve][uno_es-ve+S]	Dígito 1
[D-dos_es-ve][dos_es-ve+S]	Dígito 2
[D-tres_es-ve][tres_es-ve+S]	Dígito 3
[D-cuatro_es-ve][cuatro_es-ve+S]	Dígito 4
[D-cinco_es-ve][cinco_es-ve+S]	Dígito 5
[D-seis_es-ve][seis_es-ve+S]	Dígito 6
[D-siete_es-ve][siete_es-ve+S]	Dígito 7
[D-ocho_es-ve][ocho_es-ve+S]	Dígito 8
[D-nueve_es-ve][nueve_es-ve+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-cero_es-ve][cero_es-ve+D]	Dígito 0
[D-uno_es-ve][uno_es-ve+D]	Dígito 1
[D-dos_es-ve][dos_es-ve+D]	Dígito 2
[D-tres_es-ve][tres_es-ve+D]	Dígito 3
[D-cuatro_es-ve][cuatro_es-ve+D]	Dígito 4
[D-cinco_es-ve][cinco_es-ve+D]	Dígito 5
[D-seis_es-ve][seis_es-ve+D]	Dígito 6
[D-siete_es-ve][siete_es-ve+D]	Dígito 7
[D-ocho_es-ve][ocho_es-ve+D]	Dígito 8
[D-nueve_es-ve][nueve_es-ve+D]	Dígito 9

Tabla 4.13. Modelos específicos de palabras para francés

<i>Palabras</i>	
[oui_fr]	Modela la palabra "sí"

[non_fr]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_fr][zero_fr+S]	Dígito 0
[S-un_fr][un_fr+S]	Dígito 1
[S-deux_fr][deux_fr+S]	Dígito 2
[S-trois_fr][trois_fr+S]	Dígito 3
[S-quatre_fr][quatre_fr+S]	Dígito 4
[S-cinq_fr][cinq_fr+S]	Dígito 5
[S-six_fr][six_fr+S]	Dígito 6
[S-sept_fr][sept_fr+S]	Dígito 7
[S-huit_fr][huit_fr+S]	Dígito 8
[S-neuf_fr][neuf_fr+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_fr][zero_fr+D]	Dígito 0
[S-un_fr][un_fr+D]	Dígito 1
[S-deux_fr][deux_fr+D]	Dígito 2
[S-trois_fr][trois_fr+D]	Dígito 3
[S-quatre_fr][quatre_fr+D]	Dígito 4
[S-cinq_fr][cinq_fr+D]	Dígito 5
[S-six_fr][six_fr+D]	Dígito 6
[S-sept_fr][sept_fr+D]	Dígito 7
[S-huit_fr][huit_fr+D]	Dígito 8
[S-neuf_fr][neuf_fr+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_fr][zero_fr+S]	Dígito 0
[D-un_fr][un_fr+S]	Dígito 1
[D-deux_fr][deux_fr+S]	Dígito 2
[D-trois_fr][trois_fr+S]	Dígito 3
[D-quatre_fr][quatre_fr+S]	Dígito 4
[D-cinq_fr][cinq_fr+S]	Dígito 5
[D-six_fr][six_fr+S]	Dígito 6
[D-sept_fr][sept_fr+S]	Dígito 7
[D-huit_fr][huit_fr+S]	Dígito 8
[D-neuf_fr][neuf_fr+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	
[D-zero_fr][zero_fr+D]	Dígito 0
[D-un_fr][un_fr+D]	Dígito 1
[D-deux_fr][deux_fr+D]	Dígito 2
[D-trois_fr][trois_fr+D]	Dígito 3
[D-quatre_fr][quatre_fr+D]	Dígito 4
[D-cinq_fr][cinq_fr+D]	Dígito 5
[D-six_fr][six_fr+D]	Dígito 6
[D-sept_fr][sept_fr+D]	Dígito 7
[D-huit_fr][huit_fr+D]	Dígito 8
[D-neuf_fr][neuf_fr+D]	Dígito 9

Tabla 4.14. Modelos específicos de palabras para inglés americano

<i>Palabras</i>	
[yes_en-us]	Modela la palabra "sí"
[yeah_en-us]	Modela la palabra "sí"
[sure_en-us]	Modela la palabra "sí"
[no_en-us]	Modela la palabra "no"
[not_en-us]	Modela la palabra "no"
<i>Dígitos en contexto silencio (silencio anterior y posterior)</i>	
[S-zero_en-us][zero_en-us+S]	Dígito 0
[S-oh_en-us][oh_en-us+S]	Dígito 0
[S-one_en-us][one_en-us+S]	Dígito 1
[S-two_en-us][two_en-us+S]	Dígito 2
[S-three_en-us][three_en-us+S]	Dígito 3
[S-four_en-us][four_en-us+S]	Dígito 4
[S-five_en-us][five_en-us+S]	Dígito 5
[S-six_en-us][six_en-us+S]	Dígito 6
[S-seven_en-us][seven_en-us+S]	Dígito 7
[S-eight_en-us][eight_en-us+S]	Dígito 8
[S-nine_en-us][nine_en-us+S]	Dígito 9
<i>Dígitos en contexto silencio anterior y dígito posterior</i>	
[S-zero_en-us][zero_en-us+D]	Dígito 0
[S-oh_en-us][oh_en-us+D]	Dígito 0
[S-one_en-us][one_en-us+D]	Dígito 1
[S-two_en-us][two_en-us+D]	Dígito 2
[S-three_en-us][three_en-us+D]	Dígito 3
[S-four_en-us][four_en-us+D]	Dígito 4
[S-five_en-us][five_en-us+D]	Dígito 5
[S-six_en-us][six_en-us+D]	Dígito 6
[S-seven_en-us][seven_en-us+D]	Dígito 7
[S-eight_en-us][eight_en-us+D]	Dígito 8
[S-nine_en-us][nine_en-us+D]	Dígito 9
<i>Dígitos en contexto dígito anterior y silencio posterior</i>	
[D-zero_en-us][zero_en-us+S]	Dígito 0
[D-oh_en-us][oh_en-us+S]	Dígito 0
[D-one_en-us][one_en-us+S]	Dígito 1
[D-two_en-us][two_en-us+S]	Dígito 2
[D-three_en-us][three_en-us+S]	Dígito 3
[D-four_en-us][four_en-us+S]	Dígito 4
[D-five_en-us][five_en-us+S]	Dígito 5
[D-six_en-us][six_en-us+S]	Dígito 6
[D-seven_en-us][seven_en-us+S]	Dígito 7
[D-eight_en-us][eight_en-us+S]	Dígito 8
[D-nine_en-us][nine_en-us+S]	Dígito 9
<i>Dígitos en contexto dígito (dígito anterior y posterior)</i>	

[D-zero_en-us][zero_en-us+D]	Dígito 0
[D-oh_en-us][oh_en-us+D]	Dígito 0
[D-one_en-us][one_en-us+D]	Dígito 1
[D-two_en-us][two_en-us+D]	Dígito 2
[D-three_en-us][three_en-us+D]	Dígito 3
[D-four_en-us][four_en-us+D]	Dígito 4
[D-five_en-us][five_en-us+D]	Dígito 5
[D-six_en-us][six_en-us+D]	Dígito 6
[D-seven_en-us][seven_en-us+D]	Dígito 7
[D-eight_en-us][eight_en-us+D]	Dígito 8
[D-nine_en-us][nine_en-us+D]	Dígito 9

Atención

En gramáticas ABNF los modelos deben ir entrecomillados (véase el ejemplo anterior), mientras que en el caso de vocabularios convencionales se escriben sin necesidad de delimitarlos entre comillas.

En aplicaciones muy concretas o en aquellas que deban operar en condiciones ruidosas, etc. pueden entrenarse modelos específicos de las locuciones a reconocer (incluso para cada operario, etc), integrándose dentro de *Verbio* de forma automática. De este modo, se consigue un aumento muy notable de las tasas de reconocimiento en el entorno de trabajo específico o bajo el uso del sistema por parte de los usuarios entrenados o ambos.

4. Uso de gramáticas

Tal y como se ha descrito en el anterior apartado, el reconocedor necesita disponer de una gramática de reconocimiento o un modelo estadístico que describa el lenguaje permitido en un determinado instante del diálogo.

4.1. Crear una gramática

Las gramáticas de reconocimiento son ficheros de texto que pueden editarse desde cualquier editor. Para las gramáticas ABNF existe la herramienta *Verbio Grammar Manager* que permite no sólo editar sino compilar, probar con cadenas de texto o con voz (micrófono), generar frases permitidas por la gramática, etc. Para más información acerca de *Verbio Grammar Manager* puede consultar el documento *Verbio Software Reference: Guía del Usuario de Grammar Manager*.

La Sección 3 detalla el formato correcto de las gramáticas a usar en *Verbio*. Los modelos estadísticos serán creados por personal de *Verbio*

4.2. Preparar el vocabulario

Una vez se ha creado una gramática, ésta tiene que prepararse para su uso en el sistema de reconocimiento de *Verbio*. Preparar (o compilar) una gramática supone procesar el fichero de gramática para obtener la estructura del lenguaje (palabras aisladas o conectadas si se especifica o la estructura de nodos compleja en el caso de las gramáticas ABNF) y extraer un vocabulario de palabras que se debe transcribir a fonemas reconocidos por el sistema.

Por lo tanto se pueden producir errores de dos tipos: a) de estructura, por ejemplo una regla usa una referencia no especificada en la gramática, o b) de fonética, donde una palabra no tiene transcripción estándar en el idioma especificado. En el primer caso la herramienta de edición de gramáticas *Verbio Grammar Manager* permite identificar con más facilidad estos errores. En el segundo caso, el sistema indica la palabra que ocasiona problemas de transcripción. La solución pasa por escribir correctamente la palabra o una forma equivalente en el idioma de interés en el caso de estrangerismos o acrónimos.

Importante

Es necesario preparar el vocabulario cada vez que cambia la versión de *Verbio*, puesto que el modelado fonético puede haber cambiado. Igualmente, sólo es necesario preparar el vocabulario una vez (siempre que no sea un vocabulario dinámico, canviante con el tiempo) para todos los recursos que lo utilicen,

por lo que se recomienda que este proceso se realice, cuando sea posible, *off-line* o al arrancar una aplicación (consume muchos recursos *hardware*). En caso de que sea un vocabulario dinámico, debe tenerse la precaución de prepararlo cada vez que se modifique, con las funciones adecuadas (asociadas al canal), para evitar conflictos entre preparaciones “paralelas” en distintos canales simultáneamente.

Las gramáticas *built-in* incorporadas en *Verbio*, así como los modelos estadísticos del lenguaje (SLM) no deben prepararse, puesto que ya lo vienen por defecto dentro del propio motor de reconocimiento.

4.3. Activar una gramática para el reconocimiento

Una vez se dispone de una gramática de reconocimiento preparada para su uso en *Verbio*, se tiene que activar para que puedan reconocerse elocuciones en base a ésta. El proceso de activación se divide en un proceso de carga (*load*) en memoria y una activación (*activate*). A la inversa existen los procesos de desactivación (*deactivate*) y descarga (*unload*).

La carga en memoria reduce la carga computacional a la hora de activar/desactivar de forma dinámica algunas gramáticas.

Importante

Es imprescindible que al menos una gramática esté activa antes de intentar reconocer una elocución.

4.3.1. Activar múltiples gramáticas

El sistema de reconocimiento necesita al menos una gramática activa para poder reconocer, lo que no implica que no pueden coexistir varias simultáneamente. Es decir, *Verbio* permite que se activen simultáneamente dos o más gramáticas, lo cual es útil cuando se espera una respuesta del usuario (por ejemplo una fecha) pero éste tiene la opción de responder con una opción alternativa (por ejemplo un menú tipo “repetir”, “inicio”, etc). La pseudo-gramática activa resultante será la opcional que permite reconocer cualquiera de las expresiones que contemplaban las gramáticas activas de forma independiente.

Verbio permite conocer a cuál de las gramáticas activas pertenece la cadena de texto reconocida por el sistema.

5. Reconocer una elocución

Como se menciona en la Sección 4, es imprescindible que exista una o varias gramáticas activas antes de iniciar el proceso de reconocer una elocución.

5.1. Activar el reconocimiento de habla

Al activar el reconocimiento de una elocución deben indicarse las condiciones de terminación de este proceso. Las condiciones son de tiempo y/o eventos externos a la voz, como por ejemplo que el usuario pulse una tecla del teléfono (detección de tonos DTMF).

Los condicionantes de tiempo utilizados en *Verbio* para el reconocimiento son básicamente tres:

<code>Timeout (initsil)</code>	Es el tiempo máximo de silencio permitido desde que se activa el reconocimiento hasta que el usuario empieza a hablar. Por lo tanto, es el silencio inicial o el tiempo que el sistema permite hablar al usuario antes de terminar y decidir que el usuario no responde. Este parámetro se indica con la variable <code>initsil</code> en los distintos SDK.
<code>Complete timeout (maxsil)</code>	Es el tiempo máximo de silencio permitido una vez ya se ha detectado voz por parte del usuario. Por lo tanto es el tiempo de silencio desde que el usuario ha dejado de hablar hasta que el sistema decide que termina el reconocimiento. Consecuentemente, también es el silencio máximo permitido entre palabras. Es importante configurar de forma adecuada este parámetro en función del tipo de la gramática. Este parámetro se indica con la variable <code>maxsil</code> en los distintos SDK.
<code>Maxspeech timeout (maxtime)</code>	Es el tiempo máximo de una elocución, y comprende tanto voz como silencio. Es decir, independientemente del uso de silencio y voz es el tiempo máximo en que está reconociendo. Este parámetro se indica con la variable <code>maxtime</code> en los distintos SDK.

A las anteriores condiciones de terminación debe de añadirse las condiciones especiales de terminación que son dependientes de la plataforma. En el caso de las aplicaciones de telefonía, debe considerarse el uso del teclado telefónico, puesto que en muchos casos se ofrece al usuario la posibilidad de usar la voz o el teclado DTMF para responder a determinadas preguntas. En este caso el uso del teclado implica el no uso de la voz como respuesta por parte del usuario.

La activación del reconocimiento puede conllevar, dependiendo del SDK y la plataforma, la activación opcional del pitido que indica que empieza a reconocer.

5.2. Formato de las muestras de voz

La activación del reconocimiento supone empezar a enviar muestras de voz de la plataforma de adquisición de audio al reconocedor. Debe indicársele al reconocedor el formato en que se enviarán estas muestras. Esto es, frecuencia de muestreo (por defecto y en aplicaciones telefónicas 8 KHz) y formato de muestras (lineal con 16 bits, 8 bits en ley A o en ley U).

5.3. Barge-in

En plataformas que lo permiten (por ejemplo tarjetas Dialogic con CSP que efectúe la cancelación de eco) el reconocimiento puede funcionar de forma simultánea a la reproducción de audio proveniente de ficheros o del conversor de texto en habla. *Verbio* proporciona funciones para activar de forma conjunta reproducción de audio y reconocimiento del habla que interrumpa (barge-in) esta reproducción, lo que implica que el usuario puede hablar antes de que finalice la reproducción de audio (la pregunta) por parte del sistema.

6. Obtener el resultado

6.1. Medidas de confianza

La verificación del reconocimiento (es decir, asignar una medida de confianza al resultado de un reconocimiento) es una tarea tanto o más compleja que el propio reconocimiento. Esta verificación se basa en comparar las probabilidades de dos sistemas de reconocimiento independientes ante la misma elocución de entrada. Como resultado se obtiene un *score* o valor no acotado que indica si el reconocimiento obtenido es “fiable” y, por lo tanto, permite rechazar algunos resultados de reconocimiento, o bien sugiere confirmarlos en función de la tolerancia al error de la aplicación.

Por lo general, los valores de *score* oscilan entre 0 y 40 para resultados poco fiables y valores superiores a 50 para resultados fiables o con un índice de confianza alto. Además, estos valores son dependientes del vocabulario utilizado y por lo tanto los umbrales de aceptación y rechazo deben de ajustarse en función de la gramática y su vocabulario para afinar la precisión en el reconocimiento. Por lo tanto, es aconsejable, durante la fase de desarrollo y pruebas, realizar un estudio de los valores habituales y establecer, para cada caso, un umbral de aceptación y otro de rechazo. Se entiende por umbral de aceptación aquél por encima del cual se acepta el resultado devuelto por el reconocedor. Se entiende por umbral de rechazo aquél por debajo del cual no se acepta el resultado y, por lo tanto, debe repetirse la pregunta. En caso de que el *score* esté entre ambos umbrales, es recomendable realizar una confirmación explícita (sí/no) del resultado reconocido.

6.2. Interpretación semántica

El resultado del reconocimiento puede consultarse a nivel de palabras reconocidas, obteniendo el número de palabras reconocidas y cada una de las palabras junto con su medida de confianza e intervalos de tiempo donde se ha detectado la palabra.

Cuando se usan gramáticas ABNF existen funciones de procesamiento del resultado que permiten la interpretación semántica del resultado en base al formato utilizado al escribir la gramática. Por ejemplo, para consultar las palabras reconocidas y pertenecientes a una regla específica de la gramática, saber si el usuario ha rellenado un campo en particular, etc. Además, se puede obtener una medida de confianza para la regla especificada y los instantes de tiempo en que se ha detectado dicha parte de la elocución.

Este resultado se obtiene en los distintos SDK en una cadena de texto a procesar por parte del programador o bien es consultable mediante funciones específicas de la clase *VerbioResult* en la *Advanced SDK* (consulte la documentación correspondiente).

6.3. Múltiples hipótesis

En algunas aplicaciones resulta interesante poder disponer no sólo de la secuencia de palabras más probable sino también de otras hipótesis acerca de la elocución pronunciada. Un tratamiento posterior del conjunto de secuencias más probables (en función de su longitud, estructura, códigos de control de errores o checksum, si se encuentran en una base de datos, etc) puede permitir seleccionar con gran confianza la secuencia reconocida.

Por ejemplo, si al reconocer una secuencia de dígitos de un número de teléfono las hipótesis más probables son 994506789, 934506789 y 9345067899, podemos descartar la primera y la última, por no corresponder a un número válido de teléfono. Análogamente puede utilizarse esta técnica en el reconocimiento de códigos con dígitos o letras de control como las tarjetas de crédito, los números de identificación personal españoles (NIF), etc.

El número máximo de hipótesis a devolver por el reconocedor debe especificarse antes de activar el reconocimiento con el parámetro *NBest*. Sin embargo, como se documenta en los distintos SDK, al establecer esta valor se descargan todas las gramáticas activas en el reconocedor. Por lo tanto, en función de las gramáticas que debe usar la aplicación a lo largo de toda la ejecución, es recomendable fijar este valor al principio de la misma, de modo que en los sucesivos reconocimientos se fijará el número de hipótesis deseadas mediante el parámetro *ActiveNBest*. El valor de *ActiveNBest*, por lo tanto, siempre estará comprendido entre 1 y *NBest*.

Capítulo 5. Conversión de texto en habla

La conversión de texto en habla permite que una aplicación tenga la funcionalidad de transformar un texto escrito en habla. Es decir, permite que la máquina pueda “leer en voz alta” cualquier texto.

Para ello utiliza una expansión inteligente de números, fechas, abreviaturas, etc., que puede personalizarse mediante diccionarios propios. Mediante un lenguaje estándar de marcas se puede controlar el comportamiento del conversor, para cambiar el idioma o el locutor, variar la velocidad de elocución, introducir pausas, solicitar notificaciones, etc.

Este capítulo explica cómo trabajar con el conversor de texto en habla de *Verbio* y detalla los conceptos y pasos más importantes involucrados.

1. Inicialización

La inicialización del motor de conversión de texto en habla consiste en habilitar uno o más idiomas y locutores para usar en la aplicación así como especificar el idioma y locutor por defecto (ver funciones de inicialización en la Function Reference para más detalles o la Sección 6.3).

2. Conceptos básicos

La conversión de texto en habla es totalmente automática y por ello se basa única y exclusivamente en el texto a convertir. Es por lo tanto sumamente importante que el texto esté bien escrito, poniendo especial atención en la puntuación, los acentos y errores tipográficos y/o ortográficos. Nótese que aunque una persona es capaz de leer sin problemas el texto escrito “el 9 de octrbe”, no será así probablemente con la voz resultante de sintetizar dicho texto.

En general, es preferible facilitar en la medida de lo posible la tarea de conversión de texto en habla. Por ejemplo: introducir signos de puntuación adecuados cuando se conoce el texto a sintetizar (no es así por ejemplo en la lectura de correos electrónicos), desambiguar abreviaturas y excepciones (expandiéndolas o incorporando diccionarios personalizados al contexto de uso previsto), estandarizar fechas, cantidades monetarias o cualquier tipo de texto no normalizado sobre el que se tenga control.

Verbio proporciona la herramienta *Verbio Read Aloud*, que permite probar todos los locutores incorporados en *Verbio*, así como el comportamiento de los mismos ante cambios en la configuración, diccionarios, etc. Para más información de esta herramienta, puede consultar la documentación asociada *Verbio Software Reference: Guía del Usuario de Read Aloud*.

3. Marcación de síntesis del habla

3.1. Lenguajes de marcación de síntesis del habla

Los lenguajes de marcación para los sistemas de conversión de texto en habla surgen como herramienta para añadir información adicional a un texto a sintetizar. Permiten especificar conceptos paralingüísticos (desde velocidad de elocución a sexo del locutor) o de interpretación del texto en segmentos ambiguos, como por ejemplo delimitar que un fragmento de texto se corresponde con una fecha.

3.1.1. SSML

El consorcio W3C (<http://www.w3.org/TR/speech-synthesis/>) desarrolla un estándar de marcación basado en XML llamado *Speech Synthesis Markup Language (SSML)*. A continuación se muestran las etiquetas soportados por *Verbio*. Para información adicional, consulte la página web de W3C.

Etiqueta	Atributos	Valores permitidos
p, paragraph	xml:lang	código del idioma
say-as	interpret-as	"date", "time", "literal", "telephone", "currency", "cardinal", "ordinal", "digits"
	format	"dmy", "mdy", "ymd", "dm", "md", "my", "ym", "d", "m", "y", "hms", "hm", "ms", "h", "m", "s"
	detail	<i>atributo no soportado</i>
sub	alias	texto a utilizar para la sustitución
voice	xml:lang	código del idioma
	gender	"male", "female"
	age	entero representando la edad del locutor
	variant	<i>atributo no soportado</i>
	name	nombre de la voz a utilizar
emphasis	level	"strong", "moderate", "none", "reduced"
prosody	pitch	"x-high", "high", "medium", "low", "x-low", "default" o bien especificar un valor absoluto (p. ej.: "100Hz") o un cambio relativo (p. ej.: "-20%", "+10Hz", "-.5st")
	contour	<i>atributo no soportado</i>
	range	"x-high", "high", "medium", "low", "x-low", "default" o bien especificar un valor absoluto (p. ej.: "40Hz") o un cambio relativo (p. ej.: "-20%", "+10Hz", "-.5st")
	rate	"x-fast", "fast", "medium", "slow", "x-slow", "default" o bien especificar un valor absoluto (p. ej.: "180") o un cambio relativo, por ejemplo: (p. ej.: "+10%", "+20")
	duration	<i>atributo no soportado</i>
	volume	"silent", "x-soft", "soft", "medium", "loud", "x-loud", "default" o bien especificar un valor absoluto (entre 0 y 100) o un cambio relativo, (p. ej.: "-20%", "+20")
break	strength	none, "x-small", "small", "medium", "large", "x-large"
	time	duración de la pausa, por ejemplo: "250ms", "3s"
audio	src	URI del documento
desc	-	-
mark	name	identificador de la marca

Listado de las etiquetas de SSML soportados por *Verbio*.

La herramienta *Verbio Read Aloud* permite modificar el comportamiento del sintetizador del habla *Verbio* mediante las etiquetas SSML aquí mostradas.

En general conviene saber que no es muy aconsejable usar valores extremos para modificar las locuciones, aunque se pueden utilizar puntualmente. Si queremos aumentar la duración de los mensajes, es conveniente hacerlo progresivamente empezando por los valores por defecto, hasta llegar a un valor satisfactorio que no añada ruidos en la señal. Si empezamos por valores muy altos o muy bajos, es seguro que incorporaremos ruidos y desequilibrios en la locución. Por el mismo motivo, puede resultar más eficaz combinar el uso moderado de varias opciones de configuración que no forzar mucho una única propiedad. Para resaltar una palabra no usaremos un único parámetro (por ejemplo subir mucho la energía de esa palabra), será más eficiente hacer un poco más lenta la locución, aumentar moderadamente la energía, y rodear la palabra de pausas cortas.

Combinando las diferentes etiquetas se pueden mejorar considerablemente las locuciones generadas. Aún así, para conservar la calidad de la locución, se deben mantener los parámetros dentro del rango de naturalidad de cada locutor sin pretender cambiar sus características intrínsecas. Pretender que un locutor masculino use frecuencias superiores a 300 Hz, o que cualquier locutor hable a velocidades muy extremas siempre introduce ruidos en la señal. El uso de las etiquetas y los resultados que obtendremos nos orientarán sobre lo moldeable que es cada locutor.

3.1.1.1. Añadir pausas

```
<break strength="x-weak">
<break strength="weak">
<break strength="medium">
<break strength="strong">
<break strength="x-strong">
<break time="300ms">
```

Con estas etiquetas podemos precisar la duración de las pausas que introducimos de manera precisa y coherente.

Con la introducción de pausas breves se mejoran aspectos de entonación y naturalidad de la locución. Según situemos las pausas, podemos llegar a facilitar la comprensión de algunas locuciones:

Por favor, diga el número que encontrará <break strength="x-weak"> en la parte superior de la factura.

Introducir pausas es a menudo una de las soluciones más rápidas, fáciles y eficientes para mejorar una locución, pues cambia la entonación y la selección de unidades de síntesis, dando lugar a una oración totalmente nueva.

Utilizar pausas de diferentes duraciones puede aumentar la naturalidad de las locuciones. Las pausas más largas resaltan una menor relación entre los elementos separados. En cambio, una pausa corta puede resaltar la unidad sintáctica de los elementos que están entre pausas.

3.1.1.2. Énfasis pre-configurados

```
<emphasis level="reduced">hola</emphasis>
<emphasis level="moderate">hola</emphasis>
<emphasis level="strong">hola</emphasis>
```

Esta opción permite realizar un texto o parte del texto de manera sencilla, rápida y coherente.

El uso más común de esta etiqueta es el de destacar una parte del texto que deseamos tenga una especial importancia y claridad. Por ejemplo, son muy comunes los mensajes en que se desea focalizar la atención en alguna palabra del mensaje:

```
Si desea ser atendido en castellano, diga <emphasis level="moderate"> castellano. </emphasis>
```

Otras veces, será interesante destacar oraciones enteras:

```
Ha elegido enviar el mensaje de voz a la papelera. <emphasis level="moderate">
Recuerde que el contenido de la papelera se elimina cada media noche. </emphasis>
```

La etiqueta `emphasis level` tiene tres niveles de enfatización preconfigurados. Cada uno de estos niveles tiene predefinidas opciones lingüísticas de velocidad de locución, rango de entonación y energía de enunciación. De esta manera el usuario solo tiene que escoger en cada ocasión qué nivel de realzado desea dar al texto, sin tener que preocuparse por valores prosódicos que, en ocasiones, pueden quedar lejos del interés del usuario.

Los modelos predefinidos tienen la virtud de poder aplicar de manera extremadamente fácil las mismas especificaciones prosódicas en diferentes mensajes, sin tener que recordar ni consultar las modificaciones prosódicas concretas que se aplicaron en otras ocasiones.

3.1.1.3. Configuración del tono

```
<prosody pitch="x-low">hola</prosody>
<prosody pitch="low">hola</prosody>
<prosody pitch="medium">hola</prosody>
<prosody pitch="high">hola</prosody>
<prosody pitch="x-high">hola</prosody>
<prosody pitch="+92%">hola</prosody>
```

Esta opción permite especificar con qué tono debe pronunciar el locutor un texto o parte del texto. Por ejemplo, si toda una oración tiene una entonación más alta que la deseada, podemos bajar esta entonación manteniendo la curva melódica pero reproduciéndola a un tono más bajo.

```
<prosody pitch="low"> Ha elegido guardar el mensaje de voz. </prosody>
```

En cambio, si utilizamos la etiqueta `prosody pitch` en solo una parte de la oración estaremos modificando la curva entonativa. Por ejemplo, si deseamos realzar el carácter interrogativo de una frase, podemos utilizar esta etiqueta en la palabra final del texto a sintetizar.

```
¿Desea eliminar el mensaje <prosody pitch="high"> mensaje? </prosody>
```

La etiqueta `prosody pitch` tiene cinco niveles preconfigurados. Si deseamos mayor precisión que la que nos ofrecen las etiquetas predefinidas, podemos recurrir a especificar el incremento tonal manualmente. Esta etiqueta permite forzar incrementos o reducir porcentualmente el tono de la locución. De este modo, tenemos una infinidad de posibilidades para establecer la entonación que más deseemos.

3.1.1.4. Configuración del rango tonal

```
<prosody range="x-low">hola</prosody>
<prosody range="low">hola</prosody>
<prosody range="medium">hola</prosody>
<prosody range="high">hola</prosody>
<prosody range="x-high">hola</prosody>
<prosody range="+30%">hola</prosody>
```

Esta opción permite especificar la variabilidad tonal dentro de un grupo entonativo. Dicho llanamente, significa que podremos decidir si deseamos una entonación más plana o más dinámica. Si interesa generar una entonación plana deberemos especificar:

```
<prosody range="low"> Esta entonación es plana. </prosody>
```

Así obtendremos una entonación muy estable, poco variable y sin estridencias. Si por el contrario, deseamos una entonación más dinámica y expresiva, que tenga fragmentos con tonos altos y bajos, deberemos usar la etiqueta de la siguiente forma:

```
<prosody range="x-high"> Esta entonación es más variable. </prosody>
```

Están a su disposición otras tres opciones intermedias y aún podemos ajustar manualmente el aumento o disminución del rango tonal, hasta llegar a la amplitud deseada de la curva entonativa.

3.1.1.5. Configuración de la velocidad

```
<prosody rate="x-slow">hola</prosody>
<prosody rate="slow">hola</prosody>
<prosody rate="medium">hola</prosody>
<prosody rate="fast">hola</prosody>
<prosody rate="x-fast">hola</prosody>
<prosody rate="210">hola</prosody>
```

Esta opción permite determinar el número de palabras por minuto que realiza el locutor. En otras palabras, permite escoger la velocidad de elocución. Ralentizar una oración o parte de la oración, creará una sensación de claridad y serenidad en el mensaje:

```
Si desea ayuda diga: <prosody rate="slow"> ayuda. </prosody>
```

Por el contrario, aumentar la velocidad de elocución creará una sensación de dinamismo y puede agilizar partes de menor relevancia en un portal de voz:

```
Gracias por responder a nuestra encuesta de consumo.
<prosody rate="fast"> En cumplimiento de lo establecido en el artículo 5 de la Ley Orgánica de Protección de Datos,
VERBIO le informa que los datos que ha puesto en su conocimiento, pasarán a formar parte de un fichero automatizado
de su titularidad para la consecución de sus finalidades legítimas. </prosody>
```

Los cambios de velocidad de elocución pueden usarse también para distinguir partes del texto a sintetizar. Por ejemplo:

```
<prosody rate="fast"> El mensaje recibido ayer a las diez y media de la mañana tiene el siguiente texto: </prosody>
<prosody rate="slow">Juan: Se ha aprobado el presupuesto y la operación sigue a delante. </prosody>
```

Hay cinco etiquetas predefinidas de velocidad, y también se puede especificar el número concreto de palabras por minuto que deseamos pronuncie el sistema. Por defecto los locutores están alrededor de las 180 palabras por minuto.

3.1.1.6. Configuración de la energía

```
<prosody volume="x-soft">hola</prosody>
<prosody volume="soft">hola</prosody>
<prosody volume="medium">hola</prosody>
<prosody volume="loud">hola</prosody>
<prosody volume="x-loud">hola</prosody>
<prosody volume="50">hola</prosody>
```

Esta opción permite controlar el volumen de una locución. El uso más común de esta etiqueta es el de realzar una parte del texto que deseamos tenga una especial importancia y claridad. Por ejemplo, son muy comunes los mensajes del siguiente tipo, donde se desea focalizar la atención en alguna palabra del mensaje:

```
Si desea ser atendido en castellano, diga <prosody volume="loud"> castellano. </prosody>
```

Otras veces, será interesante destacar oraciones enteras:

```
Ha elegido enviar el mensaje de voz a la papelera.
<prosody volume="moderate"> Recuerde que el contenido de la papelera se elimina cada media noche. </prosody>
```

En cualquier caso, las combinaciones con esta etiqueta siempre darán mejor resultado que usar únicamente el parámetro de la energía.

3.1.2. SABLE

SABLE es un lenguaje de marcación para sistemas de síntesis del habla que ha surgido de la combinación de 3 lenguajes previamente existentes: SSML, STML y JSML. Sin embargo, a día de hoy, su uso es de ámbito reducido ya que SSML se ha estandarizado más a través del W3C. Para información más detallada puede consultarse <http://www.research.att.com/~rws/SABPAP/sabpap.htm> y <http://www.cstr.ed.ac.uk/projects/sable/>. Las etiquetas SABLE soportadas por *Verbio* son:

- BREAK
- RATE
- VOLUME
- AUDIO
- ENGINE
- MARKER
- LANGUAGE
- SABLE
- SPEAKER
- DIV

4. Diccionarios de excepciones y abreviaturas

Verbio utiliza diccionarios de excepciones y abreviaturas para expandir correctamente palabras no normalizadas a una forma con correspondencia directa con su pronunciación. A parte de los diccionarios ya incorporados para cada idioma, *Verbio* permite establecer diccionarios para cada puerto y cambiar dinámicamente diccionarios personalizados a medida de la aplicación.

4.1. Diccionarios de excepciones de Usuario

Los diccionarios de excepciones contienen aquellas palabras cuya pronunciación no es estándar. Es decir, su transcripción fonética no cumple con las reglas estándar del idioma correspondiente. Son excepciones los neologismos y estrangerismos en general, acrónimos, etc.

Estos diccionarios tienen en cada entrada la palabra excepción (una sola palabra sin espacios) y su sustitución por una palabra (o palabras) escrita según se pronunciaría con una lectura estándar en el idioma separados por el signo igual “=” . Por ejemplo, en español:

```
hardware = "jarguar"
```

Las palabras se buscan en el diccionario primero tal cual aparecen en el texto y después en minúsculas. Es decir, si por ejemplo se encuentra la palabra “Jordi”, primero se busca “Jordi” y si no se encuentra se busca “jordi”. Por tanto, si en el diccionario se escribe

```
Jordi = "llordi"
```

leerá en pseudo-catalán la palabra “Jordi” pero no “JORDI” ni “jordi”. Por el contrario, si se escribe en el diccionario

```
jordi = "llordi"
```

se leerá en pseudo-catalán “Jordi”, “JORDI”, “jordi”, etc.

4.2. Diccionarios de abreviaturas de Usuario

Los diccionarios de abreviaturas contienen las palabras que son abreviaturas. Su formato es igual a los diccionarios de excepciones. En este caso el primer campo es la abreviatura a sustituir.

Cuando en el primer campo del diccionario de abreviaturas se incluye un punto se indica al conversor texto voz que esta abreviatura no finaliza una frase (por ejemplo, “Dr.”). Por lo tanto, si aparece un punto a continuación

de estas abreviaturas, el conversor texto a voz lo interpreta siempre como parte de la abreviatura. Si en la abreviatura no se incluye un punto se indica al conversor que, al contrario que en caso anterior, esta abreviatura si puede finalizar una frase (por ejemplo, “pta”). En este caso si aparece un punto a continuación de estas abreviaturas, el conversor texto a voz lo interpreta como parte de la abreviatura si la siguiente letra está en minúscula, o como final de frase si la siguiente letra está en mayúscula.

Los desarrolladores pueden editar el fichero de abreviaturas para adaptarlo a sus necesidades concretas. Para ello, siguiendo la sintaxi especificada en el párrafo anterior, deben modificar adecuadamente el contenido del fichero `$INSTALLDIR$\Verbio Engines\Data\TTS\lng\%idioma%\abr_%idioma%`. Una vez aplicados los cambios, es necesario reiniciar el servicio *Verbio* para que éstos tengan efecto.

Índice

C

- Componentes obligatorios, 3
- Componentes obligatorios del cliente, 4
- Componentes obligatorios del servidor, 3
- Componentes opcionales, 5
- Componentes opcionales del cliente, 6
- Componentes opcionales del servidor, 5
- Configuraciones ASR de interés, 15
- Configuraciones de reconocimiento, 5
- Consumo de los locutores, 10
- Conversión de texto en habla, 61
 - diccionario de abreviaturas, 65
 - diccionario de excepciones, 65
 - lenguajes de marcas, 61
 - Read Aloud, 26
 - SABLE, 65
 - SSML, 61

D

- Detención del servidor, 21
- Diseño de la arquitectura, 11

G

- Gramáticas, 27
 - activar, 58
 - cargar, 58
 - crear, 57
 - Grammar >Manager, 26
 - gramáticas ABNF, 29
 - gramáticas built-in
 - boolean, 33
 - code, 36
 - creditcard, 36
 - currency, 36
 - date, 34
 - digits, 34
 - nif, 37
 - number, 35, 36
 - phone, 35
 - spell, 37
 - time, 36
 - gramáticas básicas incorporadas (built-in), 32
 - gramáticas SLM, 37
 - gramáticas XML, 38
 - modelos de palabra, 38
 - preparar, compilar, 57
 - tipos, 27
 - vocabularios de listas de palabras, 28

H

- Herramientas de prueba y desarrollo, 8

I

- Inicialización del servidor, 14
- Intérprete de VoiceXML, 8

L

- Licencias, 12
- Licencias, actualización, 17
- Locutores de síntesis, 6
- Locutores TTS de interés, 16

P

- Plataformas
 - Altitude Software (Evox), 24
 - aplicaciones de escritorio, 24
 - Avaya IR (Conversant), 24
 - CTI, 23
 - Dialogic, 23
 - Eicon Diva, 23
 - Natural Microsystems, 23
 - Prosody, ProsodyS, ProsodyX, 24
- Plataformas de desarrollo de aplicaciones, 7
- Prerrequisitos de la instalación, 3
- Puesta en marcha del servidor, 21
- Puesta en marcha del servidor de LOG, 21

R

- Reconocimiento del habla, 27
 - barge-in, 59
 - entrada, 58
 - formato de muestras, 59
 - interpretación semántica, 59
 - medidas de confianza, 59
 - múltiples hipótesis, N-best, 59
 - resultados, 59
- Requisitos hardware, 9

S

- SDK
 - Altitude uCI (Evox), 26
 - Avaya IR (Conversant), 26
 - CT ADE (VOS), 26
 - Dialogic, 25, 26
 - genérica, 25, 25
 - SAPI 4/5, 26
 - WordSpotting, 25
- SSML
 - añadir pausas, 62
 - Configuración de la energía, 64
 - Configuración de la velocidad, 64
 - Configuración del rango tonal, 63
 - Configuración del tono, 63
 - Énfasis pre-configurados, 63

T

- Tamaño del vocabulario, 9
- Tuning, 20

V

- VSCM, Verbio Server Configuration Manager, 14
-