

# Revisión crítica de la literatura especializada en lenguajes controlados

## A state-of-the-art overview in controlled languages

Carlos Mario Zapata, PhD., Roberto Rosero, Ing.

Grupo de Investigación en Lenguajes Computacionales—Universidad Nacional de Colombia  
{cmzapata, rjrosero}@unalmed.edu.co

Recibido para revisión: 6 de Septiembre de 2008, Aceptado: 28 de Noviembre de 2008, Versión final: 16 de Diciembre de 2008

**Resumen**—Una de las áreas más importantes para la Lingüística Computacional es el Procesamiento del Lenguaje Natural. Algunas de las tareas que se realizan en esta área se facilitan con el uso de los denominados Lenguajes Controlados, que son subconjuntos del lenguaje natural que presentan restricciones en el vocabulario o en la estructura del lenguaje. Los lenguajes controlados se usan en escritura de documentación técnica, simplificación de los lenguajes y en Ingeniería de Software. En este artículo se realiza una revisión crítica de la literatura disponible en Lenguajes Controlados, con el fin de establecer nuevos campos de aplicación de estos lenguajes, que permitan desarrollar proyectos futuros.

**Palabras Clave**—Lenguaje controlado, Procesamiento del lenguaje natural, Lingüística computacional, Escritura de documentación técnica, Ingeniería de software.

**Abstract**—One of the most important areas of Computational Linguistics is Natural Language Processing (NLP). Some of the NLP tasks are eased by using the so-called Controlled Languages. Controlled Languages are subsets of the natural language, with constraints in vocabulary or structure. Controlled Languages are used in technical documentation writing, language simplification, and software engineering. In this paper, we make an overview of the state-of-the-art in Controlled Languages, in order to establish new application fields of them and to develop future projects in this area.

**Keywords**—Controlled language, Natural language processing, Computational linguistics, Technical documentation writing, Software engineering.

### I. INTRODUCCIÓN

El Procesamiento del Lenguaje Natural (PLN) es uno de los recursos más atractivos que hay en la Lingüística Computacional ya que, con él, es posible realizar gran cantidad de tareas que manualmente serían muy costosas en términos de tiempo y dinero. Algunas veces, se presentan textos muy grandes que deben procesar los sistemas de PLN y los resultados de sus análisis resultan un poco tediosos. Debido a esto, se introdujo el concepto de Lenguajes Controlados, los cuales facilitan el tratamiento de documentos extensos y se rigen por reglas gramaticales y léxicas establecidas según el interés de quien necesita dichos documentos [1].

Los elementos de un lenguaje controlado son iguales a los de cualquier otro lenguaje: palabras, reglas y puntuación. Un lenguaje controlado prescribe estos elementos de manera limitada (ya sea en vocabulario o estructura) y, en ocasiones, de manera formal. Los lenguajes controlados, a menudo, se componen de una terminología controlada y de una gramática. Las herramientas que los emplean se enfocan en el análisis del texto, realizando tareas de reconocimiento de patrones y análisis de la secuencia del texto para determinar si se conforma con las reglas terminológicas y sintácticas del lenguaje controlado [25] [27] [42]. Las herramientas de este tipo pueden examinar gran cantidad de características de la lengua, incluyendo sintaxis y morfología. Otras, más avanzadas, realizan análisis y revisión estilísticos y otras pueden incluir un componente que proporcione sugerencias para las expresiones alternas aprobadas.

Los Lenguajes Controlados se suelen utilizar en tareas como la escritura de documentación técnica y la simplificación de los lenguajes. Recientemente, se realizaron también aplicaciones enfocadas en la Ingeniería de Software, como la Ingeniería de Requisitos y la elaboración de consultas en bases de datos

temporales.

En este artículo se realiza un análisis crítico de la literatura disponible en Lenguajes Controlados, tomando en consideración los usos que se les suele dar, sus ventajas y desventajas. Se espera que esta revisión constituya el paso inicial para la realización de proyectos que incluyan Lenguajes Controlados.

La estructura de este artículo es la siguiente: en la Sección II se desarrolla el marco teórico de los Lenguajes Controlados; en la Sección III se hace una revisión de la literatura especializada en Lenguajes Controlados y un vistazo general de sus aplicaciones; en la Sección IV se realiza un análisis crítico acerca de las ventajas y desventajas de los Lenguajes Controlados; en la Sección V se plantean conclusiones acerca del trabajo realizado; en la Sección VI se presenta el trabajo futuro que se puede generar a raíz de esta revisión crítica de la literatura.

## II. MARCO TEORICO

Los Lenguajes Controlados nacen a partir de un incremento en la preocupación por la calidad de la documentación técnica y su traducción, ya sea que la realice un humano o un sistema informático. Un Lenguaje Controlado consta de un glosario definido y unas reglas de escritura de la documentación. Según otras definiciones, un Lenguaje Controlado es un subconjunto del lenguaje natural con sintaxis, semántica y terminología restringidas [33][43].

Un usuario que utiliza una herramienta para el procesamiento de este tipo de lenguajes espera que no sólo dicha herramienta detecte errores, sino que también proponga una expresión alterna en un Lenguaje Controlado, conforme con la expresión ingresada que no se pudo procesar de manera correcta. Algunas razones para la utilización de Lenguajes Controlados son, entre otras: la necesidad de mayor claridad y coherencia de la información, la reducción de la ambigüedad y las responsabilidades legales, la reutilización de los activos terminológicos, el incremento de la utilidad, la fácil transferencia y traducción de los documentos. Todo eso se logra con la consistencia y estandarización de la terminología, con la simplificación de la estructura de la oración y con la estandarización del formato para la creación del documento [19]. Los Lenguajes Controlados, generalmente, tienen una mayor eficacia cuando se usan en aplicaciones comerciales o industriales [44], donde la cantidad de documentos es muy alta y la información que contienen es bastante compleja. El ejemplo típico es la creación de manuales de usuario o manuales de mantenimiento [16][24][32].

La estandarización para los Lenguajes Controlados exige que todos, dentro de lo posible, contengan características similares a las siguientes:

1. Diccionario [25][45][7]: Lista de palabras adaptadas al proceso que se relaciona con el tema que se está desarrollando.
2. Opción Terminológica [25][18]: Que incluye lo siguiente:
  - Uso de la palabra seleccionada en el contexto correcto.
  - Uso solamente de las derivaciones de verbos aprobadas.
  - Identificación clara de términos técnicos.
  - Uso exclusivo de una palabra técnica como sustantivo.
  - Simplicidad en el uso de las palabras.
- 3 Longitud de las oraciones [25]: Con reglas como las siguientes:
  - Escritura de oraciones cortas (*e.g.* máximo 20 palabras).
  - Inclusión de un sólo tema por oración.
  - Uso de conjunciones coordinativas.
- 4 Instrucciones [25][18]: Tales como:
  - Uso de sólo una instrucción por oración.
  - Redacción en modo imperativo.
  - Visualización de la enumeración de las instrucciones.
  - Uso de más de una instrucción por oración, sólo si se describen acciones simultáneas.
- 5 Mensajes de Alerta [28]: Que permiten una explicación clara de los errores potenciales o sus riesgos asociados.

## III. REVISIÓN DE LA LITERATURA EN LENGUAJES CONTROLADOS Y SUS APLICACIONES

Uno de los usos iniciales de los Lenguajes Controlados lo constituye la elaboración de documentación técnica. Los lenguajes controlados son relativamente nuevos, pero tienen actualmente una gran acogida, puesto que ya no sólo se están utilizando para la realización de documentos estructurados para las empresas, sino que también este recurso viene incursionando en la Ingeniería de Software, en las telecomunicaciones, en la web y en las tecnologías digitales [14]. El uso de los lenguajes controlados es algo innovador dentro de todas las organizaciones, puesto que provoca una estandarización general al interior de ellas y facilita el entendimiento de todos los escritos que se realizan en su interior. Algunos ejemplos de estas aplicaciones se muestran a continuación y se clasifican según su uso en: lenguajes controlados para Documentación Técnica, lenguajes controlados para Simplificación del Idioma y lenguajes controlados para Ingeniería de Software

### A. Lenguajes Controlados para Documentación Técnica

- AECMA: Uno de los primeros lenguajes controlados de que se tiene noticia es el “Inglés Simplificado de AECMA” (*AECMA Simplified English*) [16]. Este lenguaje controlado se utiliza en la industria aeronáutica para hacer frente a la complejidad creciente de la documentación técnica de mantenimiento de aviones.
- GIFAS *Rationalised French*: Es un lenguaje controlado

desarrollado por el Grupo de Industrias Aeronáuticas y Espaciales de Francia, con el fin de servir de traductor inicial del AECMA al francés y para mejorar la legibilidad de las especificaciones de AECMA por parte de los lectores franceses [8].

- BTE: Una ampliación del AECMA *simplified english* es el “Inglés Técnico de Boeing” (*Boeing Technical English* o BTE), que usa Boeing para mejorar la consistencia y legibilidad de su documentación técnica [45][33].
- CTE: La empresa Caterpillar Inc., que fabrica maquinaria pesada, utiliza el llamado *Caterpillar Technical English* (CTE), un lenguaje controlado que consiste en un inventario de cerca de 70.000 términos y una gramática controlada lo suficientemente sofisticada como para escribir en esta área técnica [15][26].
- CASL: La empresa General Motors, desde 1993, inició la utilización de los lenguajes controlados para su documentación técnica [20]. El proyecto que General Motors maneja es el llamado CASL (*Controlled Automotive Service Language*), el cual es un subconjunto del inglés que consta de 62 reglas gramaticales y en el uso de términos controlados.
- Sueco controlado: Otra de las empresas automotrices que también inició el uso de los lenguajes controlados es Scania, la cual es originaria de Suecia y pretende definir una especie de “Sueco Controlado” para los documentos que se deben realizar en el mantenimiento de los camiones que fabrica [2].
- *Standard Language*: La empresa americana Ford desarrolló el *Standard Language*, como una herramienta clave para redactar las instrucciones para sus fábricas en Norteamérica [38].
- *Controlled Language at Diebold*: Este lenguaje controlado se implementó en la empresa Diebold, la cual es una compañía de servicios que proporciona soluciones de tecnología integrada, para permitir a sus clientes maximizar sus capacidades de autoservicio [34].
- PILS: Es un esfuerzo de la industria farmacéutica para estandarizar la terminología acerca de sus productos, empleando una base de conocimientos que contiene un lenguaje controlado, fácilmente traducible al inglés, el alemán y el francés [11].
- IG: En la misma área de conocimiento de la industria farmacéutica, se definió la gramática de interacción (o *Interaction Grammar*), una extensión de las DTD (*Document Type Definition*) correspondientes al lenguaje XML (*Extensible Markup Language*). Con esta extensión, se pretende unificar la información general de los documentos farmacéuticos mediante un lenguaje controlado que se puede traducir rápidamente a distintos lenguajes [12].
- *PoliceSpeak*: Es un esfuerzo para la definición de un lenguaje común para las emergencias policiales en lenguajes inglés y francés, con el fin de mejorar las comunicaciones de este tipo en el túnel que comunica Inglaterra con el continente

europeo. Este lenguaje controlado consta de una serie de protocolos de comunicación y estándares policiales [25].

#### **B. Lenguajes Controlados para Simplificación del Idioma**

- *EasyEnglish*: Se trata de un proyecto realizado por IBM para la escritura propia de los documentos que son necesarios en su empresa. Este proyecto consiste en la implementación de un lenguaje controlado y de un revisor de gramática. Este sistema, también, tiene la posibilidad de identificar posibles ambigüedades que se presenten de tipo estructural y es el usuario quien se encarga de elegir la opción que mejor se adapta para eliminar la ambigüedad. *EasyEnglish* consta de un diccionario con cerca de 80.000 palabras, pero tiene la posibilidad de permitirle al usuario la selección de un perfil de para realizar el análisis, empleando el “propio” diccionario [6][9][13].
- ACE: *Attempo Controlled English* es un lenguaje controlado que contiene un subconjunto completo del inglés estándar. Este lenguaje se diseñó para servir como especificación y representación del conocimiento. ACE permite que los usuarios expresen los textos exactamente como los redactan, en términos del dominio de su aplicación. Como cualquier lenguaje, ACE requiere entrenamiento para utilizarlo de una manera efectiva. Sin embargo, una vez un documento se escribe en este lenguaje, cualquier persona lo podrá leer y entender [17][18].
- *GM Global English*: Es un lenguaje controlado creado por General Motors Corporation, que constituye una versión genérica del CASL con el propósito de internacionalizar la comunicación [31]. En este sentido, constituye un esfuerzo superior a CASL, pues engendra una manera genérica del lenguaje que pretende tener un alto impacto en el contexto de la comunicación internacional.
- AGILE: Es un lenguaje simplificado basado en el inglés, que permite el chequeo semántico del sentido de las palabras. Debido a su componente de desambiguación, puede descubrir usos inadecuados de las palabras [23].
- *Controlled Chinese*: Es un lenguaje controlado definido para facilitar la manipulación del lenguaje chino por parte de humanos y máquinas. Este lenguaje se basa en un lexicón que se construyó tomando en consideración las diferencias en las necesidades de los humanos y los computadores [50].
- CLAT: *Controlled Language Authoring Technology* es una tecnología que permite mejorar los productos en lenguaje natural, en términos de legibilidad y comprensibilidad. De esta manera, y con la simplificación propuesta al lenguaje, se proveen fundamentos a diferentes procesos, como la traducción de máquina y la recuperación de información [39][21].
- *TransType*: Es un enfoque de Traducción de Máquina, cuyo sistema permite asistir al traductor humano por medio de una serie de sugerencias de completitud de las frases emitidas por un motor estadístico. El lenguaje subyacente a *TransType*

se debe controlar para permitir la elaboración de las sugerencias correspondientes a la traducción [22][29][36][41].

### C. Lenguajes Controlados para la Ingeniería de Software

- UN-LENCEP: Es un lenguaje controlado para la especificación de esquemas preconceptuales. Su principal uso es la captura de información relevante de un dominio particular, con el fin de realizar su traducción automática hacia los denominados esquemas preconceptuales, unos grafos especiales que permiten la representación del conocimiento. Con dichos esquemas es posible realizar la transformación automática del modelo verbal en tres de los diagramas de UML (*Unified Modeling Language*, que es el principal estándar gráfico para el desarrollo de software): los diagramas de clases, comunicación y máquina de estados [46][47][48][49].
- Lenguajes de consulta a base de datos temporales: Los sistemas temporales de las bases de datos permiten a sus usuarios almacenar información temporal combinada con información convencional. El lenguaje estándar para la realización de consultas en bases de datos convencionales es el SQL (*Structured Query Language*), pero se queda corto en presencia de bases de datos temporales. Por ello, se propusieron algunos trabajos que incorporan un lenguaje controlado para bases de datos temporales, denominado SQL/Temporal, y al cual se llega tomando como base consultas en lenguaje natural restringido [4][5][10][35].
- *Drafter*: Es un sistema para la generación de manuales de software multilingües. El *Drafter* posee una interfaz de descripción de ventanas, que hace posible el manejo de diferentes objetos en el dominio de los manuales de software, para luego realizar un proceso de planeación automática, que desemboca en la escritura de instrucciones sencillas de uso del software en un lenguaje controlado simple [37].
- *NLMenu System*: Es un sistema capaz de comprender entradas en lenguaje natural. La idea principal detrás del *NLMenu System* es el ingreso de la información mediante un conjunto de opciones de menú que cambian a medida que el usuario introduce cada parte de la información. De esta manera, el ingreso de la información en lenguaje natural se restringe a un lenguaje controlado, que es inambiguo. Además, a partir de la salida de este sistema es posible generar interfaces a bases de datos relacionales [40].
- *RealPro*: Es un sistema de generación de texto en lenguaje natural, que sirve para traducir respuestas, que pueden ser de tipo gráfico, en textos que los usuarios convencionales pueden entender. En el ámbito de la Ingeniería de Software, se usó en el proyecto LIDA (*Linguistic Assistant for Domain Analysis*), específicamente para transformar diagramas de clases en su descripción mediante sentencias en lenguaje controlado [30].
- CIRCE: Es un entorno para el procesamiento de requisitos de software en lenguaje natural. Como los requisitos se

escriben en lenguaje natural, CIRCE tiene un módulo que se encarga de convertirlos en un lenguaje controlado, que se puede analizar sintácticamente sin que pierda la esencia de la descripción en lenguaje natural [3].

### IV. ANÁLISIS CRÍTICO

En la Sección anterior, se agruparon los trabajos existentes en lenguajes controlados en tres áreas específicas: la escritura de documentación técnica, la simplificación de los idiomas y la Ingeniería de Software. Sin embargo, estas aplicaciones dan lugar a muchas otras, como la extracción de información, la generación de textos, la traducción automática, la elaboración automática de consultas e, incluso, la generación automática de esquemas conceptuales. Si bien ésta podría ser una clasificación alterna de las aplicaciones de los lenguajes controlados, se seleccionó la mencionada clasificación para poder establecer el trabajo futuro que se podría derivar de cada una de estas áreas específicas, tal como se mostrará en la Sección VI. En esta Sección, se discute una serie de ventajas y desventajas de los lenguajes controlados.

#### A. Ventajas

En cualquiera de las áreas específicas de aplicación que se mencionaron, los lenguajes controlados realizan restricciones importantes en la terminología y la estructura, lo que conlleva una reducción importante en fenómenos lingüísticos indeseables como la ambigüedad. Específicamente en la escritura de documentación técnica, la ambigüedad puede provocar malos entendidos de las instrucciones que se intenta elaborar; de allí que los diferentes dominios en que se emplean los lenguajes controlados (aeroespacial, farmacéutico, automotor, etc.) optaran por este tipo de lenguajes para mejorar la comunicación entre los diferentes individuos pertenecientes al dominio. Algo similar ocurre con las aplicaciones en Ingeniería de Software, cuyo objetivo fundamental es la eliminación de los datos irrelevantes que se puedan presentar en el discurso inicial de un usuario, para incorporar en el desarrollo de software únicamente la información más importante. En cuanto a la simplificación de los idiomas, las restricciones planteadas por los lenguajes controlados pueden tener aplicaciones potenciales en la enseñanza de dichos idiomas, sobre todo en los hablantes no nativos de una lengua.

Una vez se implementa la infraestructura de apoyo de un lenguaje controlado, los costos para la realización de las aplicaciones posteriores se reducen sustancialmente. Por ejemplo, una vez se dispone de la terminología restringida y la gramática de un lenguaje técnico, es muy fácil producir textos que cumplan con esas características, incluso de manera automática. Bajo esta óptica, los lenguajes controlados que se mencionaron para las aplicaciones de Ingeniería de Software son especialmente útiles, puesto que sus efectos en pro de la automatización de procesos son más visibles que en los otros

tipos de aplicaciones. A este respecto, se debe anotar que la utilización de los lenguajes controlados se puede dar tanto en sistemas manuales como automáticos, lo que les suministra gran versatilidad para las diversas aplicaciones en que se pueden involucrar.

Además, en cualquiera de las aplicaciones mencionadas, la reducción de la extensión de los documentos realizados bajo la estructura de un lenguaje controlado es considerable, puesto que se elimina terminología que no aporta nada al contexto en que se trabaja, la estructura de las oraciones se simplifica y se usa un vocabulario específico. En general, los documentos se hacen más cortos, pero también supremamente más específicos, pues no contienen información considerada irrelevante. También, los productos de los sistemas que emplean lenguajes controlados se hacen más claros y sencillos, de forma que cualquiera de sus usuarios los puede entender sin mucha dificultad.

### **B. Desventajas**

Una vez se restringe la terminología o la estructura de un lenguaje, cualquier modificación a esas restricciones es mucho más difícil de hacer que si el lenguaje fuera natural. Nadie piensa, por ejemplo, en una modificación de la gramática del español, puesto que la cantidad de estructuras admisibles es supremamente amplia, lo que le suministra una gran ventaja sobre cualquiera de sus subconjuntos (que son, en esencia, lenguajes controlados). En un lenguaje controlado, en cambio, la adición de nuevas restricciones la deben estudiar cuidadosamente los expertos en el uso de esos lenguajes, y no siempre son viables de implementar, puesto que pueden contradecir otras restricciones del lenguaje.

En general, los lenguajes controlados que se usan para la elaboración de documentación técnica son altamente dependientes del dominio al que se ligan. Esta desventaja es menos evidente para los lenguajes controlados enfocados en la simplificación de un idioma o en la Ingeniería de Software, puesto que se establecen dominios generales (por ejemplo la simplificación del inglés o del francés, o la elaboración de esquemas preconceptuales desde un discurso en lenguaje controlado), que no dependen, en general, del dominio específico para el cual se están definiendo. Cuando las restricciones del lenguaje controlado son más estructurales que terminológicas, la dependencia del dominio de aplicación se hace mucho menor.

En todo caso, los lenguajes controlados suelen ser difíciles de aprender y difíciles de usar. Es mucho más fácil emplear lenguaje natural, que suele ser más fluido que cualquiera de los lenguajes controlados que de él se deriven, puesto que las restricciones de uso generan menos opciones para expresar una determinada idea. Paradójicamente, la disminución de la ambigüedad trae consigo una necesidad de expresar los conceptos de manera precisa, lo cual siempre es mucho más

difícil que expresarlos de manera fluida en lenguaje natural. Por otra parte, esa dificultad de uso de los lenguajes controlados genera, también, una necesidad de entrenamiento para su uso, que es contraria al escaso entrenamiento que se requiere para el uso del lenguaje natural. A medida que los lenguajes controlados son más especializados (y menos “naturales”, en ese contexto), tienen mayores dificultades de uso y, consecuentemente, requieren mucho más capacitación. Tal es el caso de las aplicaciones de los lenguajes controlados a la Ingeniería de Software.

### **V. CONCLUSIONES**

Un lenguaje controlado es cualquier subconjunto del lenguaje natural al cual se le adicionan restricciones en la terminología o la estructura. El uso de lenguajes controlados es de gran utilidad para cualquier clase de industria ya que le permite presentar documentos de todo tipo de una manera ágil, rápida y entendible. En este artículo se presentaron aplicaciones de los lenguajes controlados a la elaboración de documentación técnica, la simplificación de los idiomas y la Ingeniería de Software.

Las principales ventajas de los lenguajes controlados radican en la reducción de la ambigüedad, la facilidad de reutilización (con la consecuente reducción de costos en las aplicaciones) y la eliminación de los detalles irrelevantes que hacen parte de una descripción en lenguaje natural. Como principales desventajas de los lenguajes controlados se pueden mencionar la rigidez de su estructura con fines de actualización, la alta dependencia del dominio (en algunos tipos de aplicaciones, como la escritura de documentación técnica), la dificultad de uso y su consecuente necesidad de entrenamiento para el uso de lenguajes controlados.

### **VI. TRABAJO FUTURO**

Existen algunas líneas de trabajo que merecen atención para la realización de proyectos futuros, en relación con los lenguajes controlados. Por ejemplo, la alta dependencia del dominio de aplicación de las aplicaciones de escritura de documentación técnica es un aspecto que se podría superar con la definición de un “metadominio”, en el cual se examinen las generalidades correspondientes a los diferentes dominios y se puedan sintetizar en un “metalenguaje” controlado. Por ejemplo, es claro que todos los dominios que emplean lenguajes controlados para la escritura de documentación técnica tienen actores y funciones que se realizan sobre algunos objetos con la mediación de ciertos equipos. El reconocimiento de esta generalidad, conduce a la definición del metalenguaje controlado. Esto mismo se puede deducir en las aplicaciones relativas a la Ingeniería de Software, por ejemplo en el caso de los lenguajes para la captura de requisitos o la elaboración de consultas en bases de datos temporales, donde ni siquiera existe

un dominio asociado, y en los cuales el “metadominio” puede ser de gran utilidad para complementar las descripciones que esté entregando un usuario en relación con el software futuro que se desea construir.

Otro de los trabajos futuros se relaciona con la generación de los textos en lenguaje controlado. En las aplicaciones de Ingeniería de Software, por ejemplo, esos textos se podrían elaborar, como en el caso de *NLMenú System*, a partir de representaciones no textuales del mundo, como por ejemplo opciones de menú. Sin embargo, también se podrían obtener a partir de diálogos que se realicen con los usuarios, de los cuales se extraiga la información de manera cuidadosa para eliminar la información irrelevante que puedan contener. Finalmente, ese diálogo podría tener un derrotero, basado en un “metadominio” definido como se mencionó en ideas previas. De este modo, se podría conducir la captura de información de manera mucho más estructurada, teniendo en mente las descripciones futuras en lenguaje controlado.

#### AGRADECIMIENTOS

Este artículo se elaboró en el marco del proyecto “Un modelo de diálogo para la generación automática de especificaciones en UN-LENCEP”, financiado por la DIME (División de Investigación de la Sede Medellín—Universidad Nacional de Colombia).

#### REFERENCIAS

- [1] Allen, J., 1995. Natural Language Understanding. Redwood City. Benjamin/Cummings.
- [2] Alqvist, I. y Sagvall Hein, A., 1996. Defining ScaniaSwedish - a Controlled Language for Truck Maintenance. Proceedings of the First International Workshop on Controlled Language Applications (CLAW96). Leuven, Belgium: Katholieke Universiteit Leuven Centre for Computational Linguistics, pp. 159-167.
- [3] Ambriola, V. y Gervasi, V. 2006. On the Systematic Analysis of Natural Language Requirements with CIRCE. Automated Software Engineering, 13:pp. 107-167.
- [4] Androutsopoulos, I., 1996. A principled Framework for Constructing Natural Language Interfaces to Temporal Databases. Ph.D. thesis, University of Edinburgh.
- [5] Angelides, C. y Englmeier, K., 2005 Collaborative design of web service networks in a multilingual user community. Springer-Verlag London Limited.
- [6] Arendse B., 1997. EasyEnglish: A Tool for Improving Document Quality. Proceedings of the fifth conference on Applied natural language processing. pp. 159-165.
- [7] Baker, K., Franz, A., Jordan, P., Mitamura, T. y Nyberg, E., 1994. Coping with Ambiguity in a LargeScale Machine Translation System. In proceedings of COLING-94, pp. 90-94.
- [8] Barthe, K., 1998. GIFTAS Rationalised French: Designing one Controlled Language to Match Another. Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98), Pittsburgh, Pennsylvania, Carnegie Mellon University, Language Technologies Institute, pp. 87-102.
- [9] Bernth, A., 1998. EasyEnglish: PreProcessing for MT. Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98). Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University. pp. 30-41.
- [10] Boehlen, M., 1996. Querying TSQL2 databases with temporal logic. In Proceedings of the 5th International Conference on extending Database Technology (EDBT), Avignon, France.
- [11] Bouayad-Agha, N., Power, R., Scott, D. y Belz, A., 2002. PILS: Multilingual generation of medical information documents with overlapping content. In Proceedings of the Third International Conference on Language resources and evaluation (LREC 2002), pp. 2111-2114.
- [12] Brun, C., Dymetman, M. y Lux, V., 2000. Document structure and multilingual authoring. In Proceedings of First International Natural Language Generation Conference (INLG 2000), pp. 24-31.
- [13] Church, K. W. y Chaplin, L. F., 1995. Commercial applications of natural language processing. Commun. ACM 38, pp. 71-79.
- [14] Danlos L., Lapalme, G. y Lux, V., 2000 Generating a Controlled Language. Annual Meeting of the ACL. Proceedings of the first international conference on Natural language generation, 14: pp. 141 – 147.
- [15] Douglas, S. y Hurs, M., 1996. Controlled English Support for Perkins Approved Clear English (PACE). Proceedings of the First International Workshop on Controlled Language Applications (CLAW96), Leuven, Belgium, Katholieke Universiteit Leuven Centre for Computational Linguistics, pp. 106-114.
- [16] Farrington, G., 1996. AECMA Simplified English: A Guide for the Preparation of Aircraft Maintenance Documentation in the International Aerospace Maintenance Language, AECMA, Brussels, Controlled Language Application Workshop (CLAW): pp. 1-21.
- [17] Fuchs, N., Hofmann, H. F. y Schwitter R., 1995. Specifying logic programs in controlled natural language. In Workshop on Computational Logic for Natural Language Processing, Proceedings CLNLP '95.
- [18] Fuchs, N. E. y Schwitter, R., 1996. Attempto Controlled English (ACE). Proceedings of the First International Workshop on Controlled Language Applications (CLAW96), Leuven, Belgium, Katholieke Universiteit Leuven Centre for Computational Linguistics, pp. 124-136.
- [19] Godden K., 1998. Controlling the Business Environment for Controlled Language. Proceedings of the Second International Workshop on Controlled Language Applications Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University, pp. 185-190.
- [20] Godden, K., 2000. The Evolution of CASL Controlled Authoring at General Motors. Proceedings of the Third International Workshop on Controlled Language Applications (CLAW 2000), Seattle, Washington, pp. 14-19.
- [21] Haller, J. y Schütz, J., 2001. CLAT: controlled language authoring technology. In Proceedings of the 19th Annual international Conference on Computer Documentation, pp. 78-82.
- [22] Hartley A., Scott, D. y Bateman, J., 2001. Translation, controlled language, generation. In E. Steiner and C. Yallop, editors, Exploring Translation and Multilingual Text production , pp. 307-325.
- [23] Holmbach H. y Harrison, A., 2000 A Word Sense Checking Application for Simplified English. Proceedings of the Third International Workshop on Controlled Language Applications Seattle, Washington: Association for Computational Linguistics, pp. 120-133.
- [24] Huijsen, W., 1998. Controlled Language - An Introduction. Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98). Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University, pp. 1-15.
- [25] Johnson, E., 2002. Talking across Frontiers: Building Communication between Emergency Services, Regional & Federal Studies, 12: pp. 88-110.
- [26] Kamprath, C., Adolphson, E., Mitamura, T. and Nyberg, E., 1998. Controlled Language for Multilingual Document Production: Experience with Caterpillar Technical English. Controlled Language Application Workshop (CLAW): pp. 1-12.
- [27] Kittredge, R., Polguere, A., 2000. The Generation of Reports from Databases. In A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text, ed. R. Dale, H. Somers, and H. Moisl. Marcel Dekker Inc.
- [28] Lalaude, M., Lux, V. y Regnier-Prost, S., 1998. Modular Controlled Language Desing. In Proceedings of the second internatioanal Workshop on Controlled Language Applications. LTI, Carnegie Mellon University.
- [29] Langlais, P., Loranger, M. Y Lapalme, G., 2002. Translators at work with TRANSTYPE: resource and evaluation. In Proceedings of the Third International Conference on Language Resources and Evaluation (LREC

- 2002), pp. 2128-2134. Las Palmas de Gran Canaria, España.
- [30] Lavoie, B. y Rambow, O., 1997. RealPro: A fast, portable sentence realizer. In Proceedings of the conference on Applied Natural Language Processing (ANLP'97), Washington, D.C.
- [31] Means, L., 2000. Training for Controlled Language Processes. In Proceedings of the third International Workshop on Controlled Language Applications (CLAW 2000), Seattle, Washington, pp. 51-61.
- [32] Mikkel T., 1994. Controlled grammatic ambiguity. ACM Transactions on Programming Languages and Systems (TOPLAS), 16: pp. 1024-1050.
- [33] Mitamura, T. y Nyberg, E. 1995. Controlled English for Knowledge-Based Machine Translation: Experience with the KANT System, In Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation (TMI).
- [34] Moore, C., 2000. Controlled Language at Diebold, Incorporated. Proceedings of the Third International Workshop on Controlled Language Applications (CLAW 2000), Seattle, Washington, pp. 51-61.
- [35] Nelken, R., et al., 2000. Querying Temporal Databases Using Controlled Natural Language. In COLING, pp. 1076-1080.
- [36] Nyberg, E. y Francez, N., 1996. Controlled Language and Knowledge-Based Machine Translation: Principles and Practice. Proceedings of the First International Workshop on Controlled Language Applications (CLAW96). Leuven, Belgium: Katholieke Universiteit Leuven Centre for Computational Linguistics. pp. 74-83.
- [37] Paris, C., Linden, K. V., Fischer, M., Hartley, A., Pemberton, L., Power, R. y Scott, D., 1995. A Support Tool for writing multilingual instructions. In proceedings of the 14th International Joint Conference on Artificial Intelligent. Montreal, Canada, pp. 1398-1404.
- [38] Rychtyckyj, N., 2002. An Assessmnet of Machine Translation for Vehicle Assembly Process Planning at Ford Motor Company. En Stephen D. Richardson (Ed.): Machine Translation: From Research to Real Users, 5th Conference of the Association for Machine Translation in the Americas, AMTA 2002 Tiburon, CA, USA. Lecture Notes in Computer Science pp. 207-215.
- [39] Schmidt-Wigger, A., 1998. Grammar and Style Checking for German. In Proceedings of CLAW, pp. 76–86, Pittsburg, PA.
- [40] Tennant, H. R. y Ross, K. M. 1983. Usable natural language interface through menu-based natural language understanding. In CHI'83 Proceedings. Computer Human Interactions.
- [41] Van der Eijck, P., De Koning, M. y Van der Steen, G., 1996. Controlled Language Correction and Translation. Proceedings of the First International Workshop on Controlled Language Applications (CLAW96). Leuven, Belgium: Katholieke Universiteit Leuven Centre for Computational Linguistics pp. 64-73.
- [42] Wojcik, R. y Harrison, P., 1990. The Boeing Simplified English Checker, Proceedings of the International Conference, Human Machine Interaction and Artificial Intelligence in Aeronautics and Space. Toulouse: Centre d'Etudes et de Recherches de Toulouse, pp. 43-57.
- [43] Wojcik, R. y Hoard, J. E., 1995. Controlled Languages in Industry. In Survey of the State of the Art in Human Language Technology, ed. Giovanni Battista Varile and Antonio Zampolli, pp. 274-276.
- [44] Wojcik, R. y Holmbach, H., 1996. Getting a Controlled Language Off the Ground at Boeing. Proceedings of the First International Workshop on Controlled Language Applications (CLAW96). Leuven, Belgium: Katholieke Universiteit Leuven Centre for Computational Linguistics, pp. 22-31.
- [45] Wojcik, R. y Holmbach, H., 1998. Boeing Technical English. An Extension of AECMA SE beyond the Aircraft Maintenance Domain. Controlled Language Application Workshop (CLAW): pp. 114-123.
- [46] Zapata, C. M., Gelbukh, A. y Arango, F., 2006. UN-Lencep: Obtención automática de diagramas UML a partir de un lenguaje controlado. Memorias del 3er Taller en Tecnologías del Lenguaje Humano del Encuentro Nacional de Computación, San Luis Potosí.
- [47] Zapata, C. M., Gelbukh, A. y Arango, F., 2006. Pre-conceptual schema: a UML isomorphism for automatically obtaining UML conceptual schemas. Research in Computing Science: Advances in Computer Science and Engineering, 19, pp. 3-13.
- [48] Zapata, C. M., Gelbukh, A. y Arango, F., 2008. UN-LENCEP: A Controlled Language for Pre-conceptual Schema Specification. Memorias de las VII Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento, Guayaquil, 2008. pp. 269-276.
- [49] Zapata, C. M., Gelbukh, A. y Arango, F., 2006. Pre-conceptual Schema: A Conceptual-Graph-Like Knowledge Representation for Requirements Elicitation. F. Lecture Notes in Computer Science, Vol. 4293, pp. 17-27.
- [50] Zhang, W., Zhou, X. y Yu, S., 1998. Construction of a Controlled Chinese Lexicon. Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98), (Pittsburgh, Pennsylvania, Carnegie Mellon University, Language Technologies Institute, pp. 159-173.



**Carlos Mario Zapata J.** es Ingeniero Civil (1991), Especialista en Gerencia de Sistemas Informáticos (1999), M. Sc. en Ingeniería de Sistemas (2002) y Ph. D. en Ingeniería (2007). Todos sus estudios los realizó en la Universidad Nacional de Colombia (Medellín, Colombia). Es Profesor Asociado de la Escuela de Sistemas de la Facultad de Minas, Universidad Nacional de Colombia. Entre sus áreas de interés se destacan la Ingeniería de Software, la Ingeniería de Requisitos, la Lingüística Computacional, el Procesamiento del Lenguaje Natural y las Estrategias Pedagógicas para la Enseñanza de la Ingeniería.

**Roberto Rosero** es Ingeniero de Sistemas e Informática de la Universidad Nacional de Colombia. Actualmente, se desempeña como Ingeniero de Desarrollo de la empresa Mantum S. A. Entre sus áreas de interés se destacan la Ingeniería de Software y la Lingüística Computacional.



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA  
SEDE MEDELLÍN  
FACULTAD DE MINAS

**120** años TRABAJO Y RECTITUD