# *Bases de Datos no Convencionales: Índices y Lenguajes de Consulta*

Jorge Arroyuelo, Susana Esquivel, Alejandro Grosso, Verónica Ludueña, Cintia Martínez, Nora Reyes
Dpto. de Informática, Fac. de Cs. Físico-Matemáticas y Naturales, Universidad Nacional de San Luis
{*bjarroyu, esquivel, agrosso, vlud, nreyes*}@unsl.edu.ar, *cintiavmartinez*@hotmail.com
Edgar Chávez
Centro de Investigación Científica y de Educación Superior de Ensenada, México
*elchavez*@cicese.mx
Karina Figueroa
Fac. de Cs. Físico-Matemáticas, Universidad Michoacana de San Nicolás de Hidalgo, México
*karina*@computo.fismat.umich.mx
Gonzalo Navarro
Dpto. de Cs. de la Computación, Universidad de Chile
*gnavarro*@dcc.uchile.cl
Manuel Hoffhein, Rodrigo Paredes
Dpto. de Cs. de la Computación, Fac. de Ingeniería, Universidad de Talca, Chile
*mahahein3*@gmail.com, *raparede*@utalca.cl

## Resumen

En la actualidad es muy común suministrar una imagen a un buscador y esperar que éste localice, imágenes parecidas a la provista. Escenarios como éste requieren el desarrollo de aplicaciones capaces de manipular datos no convencionales como imágenes, audio, video, secuencias de ADN, texto, huellas digitales, etc., almacenarlos y obtener información desde ellos, para responder eficientemente consultas que realicen los usuarios. Claramente, es necesario utilizar depósitos especializados de datos y técnicas de búsquedas no exactas sobre ellos, porque las soluciones tradicionales no permiten hacer frente a tales requerimientos. En este ámbito es muy raro comparar por igualdad exacta, siendo generalmente las consultas por objetos similares a uno dado. Por lo tanto, además de requerir una respuesta rápida y adecuada y un eficiente uso del espacio disponible, es necesario utilizar modelos generales en los cuales se puedan utilizar estructuras de datos especializadas que contemplen estos aspectos, como lo son las *Bases de Datos Métricas* y que si se consideran bases de datos masivas, dichas estructuras en particular sean, en particular, *estructuras de datos con I/O eficiente*. Otro aspecto importante son los lenguajes de consulta, necesarios para la manipulación de una base de datos, que no siempre poseen el poder expresivo necesario para expresar las consultas consideradas de interés en este modelo. Así, nuestra investigación pretende contribuir a la consolidación de este nuevo modelo de bases de datos.

***Palabras Claves:*** bases de datos no convencionales, lenguajes de consulta, índices, expresividad.

## Contexto

La actual presentación se enmarca en el Proyecto Consolidado 330314, *Tecnologías Avanzadas de Bases de Datos* de la Universidad Nacional de San Luis, dentro del Programa de Incentivos a la Investigación (Código 22/F414); particularmente en la línea *Bases de Datos no Convencionales*. Este proyecto es una nueva presentación en 2014, continuación de proyectos anteriores en la misma área. Entre las actividades centrales de esta línea situamos la investigación de aspectos teóricos, empíricos y aplicativos derivados de la administración de bases de datos que manejan tipos no convencionales de datos, útiles en distintos campos de aplicación: sistemas de información geográfica, computación móvil, diseño asistido por computadora, robótica, visión artificial, motores de búsqueda en internet, etc.. También se explora la expresividad de los lenguajes de consulta, los operadores necesarios para responder demandas de interés, y las estructuras y operaciones para resolverlas eficientemente.

La participación de nuestros integrantes en actividades de cooperación internacional con: Universidad de Chile, Universidad de Talca (Chile), Universidad Michoacana de San Nicolás de Hidalgo (México) y Centro de Investigación Científica y de Educación Superior de Ensenada (México), permite nuevas perspectivas en nuestras investigaciones.

# Introducción

Actualmente resulta muy común tratar de encontrar melodías similares a una dada en una base de datos de audio, como también ingresar una imagen a un buscador y esperar que éste localice en la WEB imágenes parecidas a la provista. Esto da lugar al desarrollo de aplicaciones totalmente diferentes a las tradicionales, que deben ser capaces de operar con nuevos tipos de datos y una filosofía de búsqueda diferente: la *búsqueda por similitud* Algunos ejemplos de aplicaciones son reconocimiento de voz, reconocimiento de imágenes, reconocimiento facial, comparación de huellas digitales, bases de datos médicas, recuperación de texto, biología computacional, clasificación y aprendizaje automático, minería de datos, etc.

Todas estas aplicaciones tienen características comunes, englobadas en el modelo de *espacio métrico*. Un espacio métrico consiste de un universo de objetos $\mathbb{U}$ y una función de distancia definida entre ellos $d : \mathbb{U} \times \mathbb{U} \longmapsto \mathbb{R}^+$ que mide la disimilitud entre los objetos. En este escenario las búsquedas exactas carecen de sentido y es importante la elección de este modelo por las *búsquedas por similitud*, más naturales sobre estos tipos de datos. En este caso se utilizan los *Métodos de Acceso Métricos* (MAMs).

La mayoría de estos métodos no admiten dinamismo, ni están diseñadas para conjuntos masivos de datos u operaciones de búsqueda complejas. Así, es posible analizar distintas maneras de optimizarlas. El trabajo con bases de datos masivas, o con aquellas que almacenan objetos muy grandes, da lugar también a líneas de investigación que, considerando el cambio del modelo de costo utilizado, diseñan MAMs más eficientes (en espacio, en I/O, etc,) para memorias jerárquicas. La obtención de mayor expresividad en los lenguajes de consulta utilizados, para expresar consultas y caracterizar la clase de consultas computables, es otra área de investigación.

# Líneas de Investigación y Desarrollo

## Lenguajes de Consulta

La relación existente entre lógica y teoría de bases de datos es muy estrecha y natural, ya que es posible pensar en una base de datos simplemente como una estructura finita, y utilizar las lógicas para expresar consultas sobre éstas. Esto les da una posición central como modelo computacional para el análisis del poder expresivo de los lenguajes de consultas, siendo relevante como marco teórico para el estudio de las bases de datos.

La mayoría de los lenguajes de consulta sobre bases de datos es equivalente, en su poder expresivo, a FO (F*irst-Order Logic*). El principal problema es que la expresividad de FO no es lo suficientemente poderosa, por lo que no alcanza para reflejar ciertas consultas. Esto ha llevado a la búsqueda de una mayor expresividad por medio de diferentes mecanismos de extensión sobre FO, utilizados como herramientas de construcción de lógicas más poderosas. Uno de estos mecanismos es incorporar cuantificadores que no pueden ser expresados en FO, como *clausura transitiva* y *punto fijo*, entre otros, los que han sido ampliamente estudiados. La idea de agregar cuantificadores se generaliza mediante la noción de *cuantificadores generalizados de Lindström* [6].

Aún así, estas lógicas todavía resultan incompletas, por lo que se analizan lógicas de orden superior. SO (*Second-Order Logic*) y algunos de sus fragmentos que han demostrado poseer propiedades interesantes sobre las estructuras finitas. Un resultado importante de R. Fagin fue la caracterización del fragmento existencial SO$\exists$ [7]. Allí se establece que las propiedades de las estructuras finitas que son definidas por sentencias existenciales de segundo orden coinciden con las propiedades de la clase de complejidad NP, lo cual fue extendido por Stockmeyer [18], estableciendo una relación cercana entre la lógica SO y la jerarquía de tiempo polinomial (PH).

Actualmente existen muchos resultados igualando la expresividad lógica a la complejidad computacional, pero requieren estructuras ordenadas [9][8]. Estas relaciones entre la complejidad computacional (cantidad de recursos necesarios para resolver un problema sobre algún modelo de máquina computacional) y la complejidad descriptiva (el orden de la lógica que se necesita para describir el problema), han llevado a que los resultados obtenidos en alguno de estos campos sea transferido de manera inmediata al otro.

En uno de nuestros trabajos de investigación definimos una nueva lógica de tercer orden (TO), la cual hemos llamado TO$^\omega$, con la idea principal de caracterizar y estudiar clases de complejidad relacionales (temporales) de lógicas de orden superior. La lógica TO$^\omega$ surge para continuar con la linea estudiada por Dawar en SO$^\omega$ que plantea una restricción semántica a la lógica de segundo orden, donde la valuación de las variables relacionales para los cuantificadores de segundo orden son cerrados bajo $\equiv_k$. Una relación

es cerrada bajo $\equiv_k$ si todas las tuplas (del dominio sobre el que trabaja) que tienen el mismo tipo están en la relación. Dos tuplas tienen el mismo tipo si satisfacen las mismas formulas de FO$^k$.

Para poder asociar TO$^\omega$ a una clase de complejidad temporal, definimos una variación de una maquina relacional no deterministica, que denotamos como 3-NRM, donde permitimos relaciones de tercer orden en el *relational store*. En base a esta maquina definimos la clase de complejidad $\mathrm{NEXPTIME}_{3,r}$, como la clase de maquinas 3-NRM que trabajan en tiempo exponencial de acuerdo al tamaño de la entrada. Luego se demostró que el fragmento existencial de TO$^\omega$ caracteriza exactamente la clase de complejidad $\mathrm{NEXPTIME}_{3,r}$. Estos trabajos fueron presentados en [1].

## Bases de Datos Métricas

Las bases de datos no convencionales se modelizarán mediante los espacios métricos. Aquí es necesario responder consultas por similitud eficientemente haciendo uso de MAMs. En espacios métricos generales la complejidad usualmente se mide como el número de cálculos de distancias realizados. Por ello, se analizan aquellos MAMs que han mostrado buen desempeño en las búsquedas, para optimizarlos más, considerando la jerarquía de memorias. En general, dada una base de datos $X \subseteq \mathbb{U}$ y una consulta $q \in \mathbb{U}$ las consultas son de dos tipos: por *rango* o de *k-vecinos más cercanos*, aunque existen otras operaciones de interés [16].

### Métodos de Acceso Métricos

A partir del *Árbol de Aproximación Espacial* [10], un índice que mostró un muy buen desempeño en espacios de mediana a alta dimensión, pero totalmente estático, se desarrolló uno de los pocos índices completamente dinámicos: el *Árbol de Aproximación Espacial Dinámico (DSAT)* [11] que permite realizar inserciones y eliminaciones, conservando su buen desempeño en las búsquedas. El *DSAT* particiona el espacio considerando la proximidad espacial; pero, si el árbol agrupara los elementos muy cercanos entre sí, lograría mejorar las búsquedas, al evitar recorrerlo. Podemos pensar entonces que construimos un *DSAT*, en el que cada nodo representa un grupo de elementos cercanos ("clusters") y los relacionamos por su proximidad en el espacio. Cada nodo mantiene el centro del cluster correspondiente, y almacena los $k$ elementos más cercanos a él; cualquier elemento a mayor distancia del centro que los $k$ almacenados, forma parte de otro nodo en el árbol [2]. Nuevas estrategias de optimización de funciones a través de heurísticas bioinspiradas, que han mostrado ser útiles en detección de clusters, pueden servir para analizar cuán bueno es el agrupamiento o "clustering"que logra esta estructura.

Dado que una base de datos métrica, ya sea por ser masiva o porque sus objetos son muy grandes, o porque el índice no quepa en memoria principal, o ambas cosas, no se almacene en memoria principal, surge la necesidad de hacer uso de la memoria secundaria. Esto requiere diseñar índices especialmente para memoria secundaria. Así, en [12] se presentaron versiones preliminares del *DSAT* (*DSAT+* y *DSAT**), que sólo admiten inserciones y búsquedas eficientes. Sin embargo, numerosas aplicaciones necesitan total dinamismo; es decir, que también puedan realizarse eliminaciones. Así, se ha diseñado un nuevo índice dinámico para memoria secundaria, basado en la *Lista de Clusters* [3], que tiene buen desempeño en espacios de alta dimensión, es completamente dinámico, con buena ocupación de página y sus operaciones son eficientes tanto en cálculos de distancia como en operaciones de I/O [13].

Sin embargo, existen otras maneras posibles de lograr un índice totalmente dinámico a partir de la *Lista de Clusters*. Por ello, estamos actualmete diseñando un nuevo índice que, a través de combinar con *algoritmos de pivotes* [3] y de considerar "clusters" cuyo tamaño se defina en función del tamaño de página de disco, logre ser eficiente en búsquedas, inserciones y eliminaciones.

En algunos casos, aunque la estructura sea eficiente, con el fin de lograr una respuesta más rápida, se intercambia precisión por velocidad en la respuesta. Es decir, se admite que ante una consulta se devuelvan sólo algunos objetos relevantes, siempre que dicha respuesta se encuentre disponible mucho más rápido. Estos tipos de búsquedas se denominan *aproximadas*. Un algoritmo muy eficiente para este tipo de consultas es el llamado *algoritmo basado en permutaciones* [4]. Por lo tanto, se está diseñando un nuevo índice que combine las ideas de [13], pero que agrupe por distancia entre las permutaciones de los objetos, en lugar de por distancia entre objetos. Esto permitiría obtener un índice al que se le pueda indicar el número máximo de cálculos de distancia y/o el número máximo de operaciones de I/O, que se está dispuesto a utilizar, para obtener una respuesta rápida; a costa de admitir que algunos objetos relevantes a la consulta no sean informados en la respuesta.

**Búsqueda Aproximada de los *All-*$k$-NN**

El modelo de *espacios métricos* abarca aplicaciones tales como la predicción de funciones: se desea buscar el comportamiento más similar de una función en el pasado para predecir su comportamiento futuro probable; la clasificación y aprendizaje automático: un nuevo elemento debe ser clasificado de acuerdo a sus vecinos más cercanos; la cuantificación y compresión de imágenes: sólo algunos vectores pueden ser representados y aquellos que no pueden serlo, deben ser codificados como su punto representable más cercano, entre otras.

Dado que, la evaluación de la función de distancia $d$ se usa como medida de complejidad en la mayoría de los casos, debido a su costo, se desarrollaron varias técnicas para resolver el problema de consultas por similitud en un número sublineal de cálculos de distancia, basadas en el *preprocesamiento* de los datos.

La recuperación de los $k-$*vecinos más cercanos* es una de las primitivas básicos de las búsquedas por similitud y puede definirse como: Sea $X$ un conjunto de elementos y $d$ la función de distancia definida entre ellos, los $k$-NN($u$) son los $k$ elementos en $X - \{u\}$ que tengan la menor distancia a $u$ de acuerdo con la función $d$. Una variante menos estudiada de este problema, es la búsqueda de los $k-$vecinos más cercanos de *todos* los elementos de $X$, *All-*$k$-NN. Sea $|X| = n$, obtener los *All-*$k$-NN es calcular los $k$-NN($u_i$) para cada $u_i$ en $X$, por supuesto realizando menos de $n^2$ cálculos de distancia. En el marco de una etapa de investigación previa, se propusieron y desarrollaron soluciones a este problema, en espacios métricos generales [15, 14], basadas en la construcción del *Grafo de los* $k$*-vecinos más cercanos* ($k$NNG). Éste indexa un espacio métrico, requiriendo una cantidad moderada de memoria, y luego se utiliza en la resolución de las consultas por similitud. El $k$NNG es un grafo dirigido ponderado que conecta cada elemento del espacio métrico mediante un conjunto de arcos cuyos pesos se calculan de acuerdo a la métrica del espacio en cuestión. El desempeño en las búsquedas por similitud de esta propuesta es superior al obtenido utilizando las técnicas clásicas basadas en pivotes.

Por otro lado, el compromiso de tratar de realizar la menor cantidad de cálculos de distancias posibles durante una búsqueda, ha llevado a investigar un enfoque *aproximado* eficiente para resolver estas consultas por similitud. Este enfoque consiste en permitir una relajación en la precisión de la respuesta a fin de obtener una aceleración en la complejidad de la de consulta [17, 3, 19]. El objetivo de la *búsqueda por similitud aproximada* es reducir significativamente los tiempos de búsqueda al permitir algunos errores en el resultado de la consulta. Además de la consulta se especifica un parámetro de precisión $\varepsilon$ para controlar cuán lejos queremos el resultado de la consulta del resultado correcto. Un comportamiento razonable para este tipo de algoritmos es acercarse asintóticamente a la respuesta correcta como $\varepsilon$ se acerca a cero. Por lo tanto, el éxito de una técnica de aproximación se basa en la resolución del compromiso calidad/tiempo [5]. Esta alternativa a la búsqueda por similitud "exacta"abarca algoritmos aproximados y probabilísticos.

## Resultados y Objetivos

Uno de los objetivos planteados es considerar los distintos aspectos relacionados al diseño de estructuras de datos que, conscientes de la jerarquía de memorias y de las características particulares de los datos a ser indexados, logren ser eficientes en espacio y en tiempo.

Por ello, se busca que los índices se adapten mejor al nivel de la jerarquía de memorias donde se almacenarán. Estos estudios, sobre espacios métricos y sobre algunas estructuras de datos particulares, permitirán no sólo mejorar el desempeño de las mismas sino también aplicar, eventualmente, muchos de los resultados que se obtengan a otros MAMs.

Respecto de los lenguajes de consulta se continuará analizando la expresividad de distintas extensiones de FO y posibles restricciones de SO, para lograr caracterizar la clase de las consultas computables sobre bases de datos no convencionales.

## Actividades de Formación

Dentro de esta línea de investigación se forman alumnos y docentes-investigadores de acuerdo al siguiente detalle:

**Doctorado en Cs. de la Computación:** un investigador de la líinea desarrolla su tesis sobre bases de datos métricas, esperando finalizarla este año. Otro integrante está realizando su tesis sobre la expresividad de la lógica como lenguaje de consulta.

**Maestría en Cs. de la Computación:** un investigador de la línea desarrolla su tesis sobre búsqueda por similitud aproximada, esperando finalizarla este año.

**Maestría en Informática**: un alumno de la Universidad Nacional de San Juan está desarrollando su te-

sis sobre un índice dinámico para búsquedas por similitud aproximadas en memoria secundaria.

**Trabajo Final de Ingeniería Civil en Computación:** un alumno de la Universidad de Talca está desarrollando su trabajo de fin de carrera sobre el diseño de un nuevo índice dinámico para memoria secundaria, basado en una combinación de técnicas de pivotes y *Lista de Clusters*, esperando finalizarlo este año.

## Referencias


[1] J. Arroyuelo and J. M. Turull Torres. The existential fragment of third order logic and third order relational machines. In *XX Congreso Argentino de Ciencias de la Computación (CACIC'14)*, pages 324 – 333, 10/2014 2014.

[2] M. Barroso, N. Reyes, and R. Paredes. Enlarging nodes to improve dynamic spatial approximation trees. In *Procs. of the 3rd International Conference on Similarity Search and Applications (SISAP)*, pages 41–48. ACM Press, 2010.

[3] E. Chávez, G. Navarro, R. Baeza-Yates, and J. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September 2001.

[4] E. Chávez, K. Figueroa, and G. Navarro. Effective proximity retrieval by ordering permutations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(9):1647–1658, Sept 2008.

[5] P. Ciaccia and M. Patella. Approximate and probabilistic methods. *SIGSPATIAL Special*, 2(2):16–19, 2010.

[6] H. Ebbinghaus and J. Flum. *Finite model theory*. Perspectives in Mathematical Logic. Springer, 1995.

[7] R. Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity of Computation, SIAM-AMS Proceedings*, 7:43–73, 1974.

[8] N. Immerman. *Descriptive Complexity*. Graduate texts in computer science. Springer New York, 1999.

[9] N. Immerman. Descriptive and computational complexity. In J. Hartmanis, editor, *Computational Complexity Theory, Proccedings of Symposia in Applied Mathematics*, volume 38, pages 75–91. American Mathematical Society, 1989.

[10] G. Navarro. Searching in metric spaces by spatial approximation. *The Very Large Databases Journal (VLDBJ)*, 11(1):28–46, 2002.

[11] G. Navarro and N. Reyes. Dynamic spatial approximation trees. *Journal of Experimental Algorithmics*, 12:1–68, 2008.

[12] G. Navarro and N. Reyes. Dynamic spatial approximation trees for massive data. In T. Skopal and P. Zezula, editors, *SISAP*, pages 81–88. IEEE Computer Society, 2009.

[13] G. Navarro and N. Reyes. Dynamic list of clusters in secondary memory. In *Proc. 7th International Workshop on Similarity Search and Applications (SISAP)*, LNCS 8821, pages 94–105, 2014.

[14] R. Paredes. *Graphs for Metric Space Searching*. PhD thesis, University of Chile, Chile, July 2008.

[15] R. Paredes, E. Chávez, K. Figueroa, and G. Navarro. Practical construction of $k$-nearest neighbor graphs in metric spaces. In *Proc. 5th Workshop on Efficient and Experimental Algorithms (WEA)*, LNCS 4007, pages 85–97, 2006.

[16] R. Paredes and N. Reyes. Solving similarity joins and range queries in metric spaces with the list of twin clusters. *Journal of Discrete Algorithms*, 7(1):18–35, 2009.

[17] H. Samet. *Foundations of Multidimensional and Metric Data Structures (The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.

[18] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.

[19] P. Zezula, G. Amato, V. Dohnal, and M. Batko. *Similarity Search: The Metric Space Approach (Advances in Database Systems)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005. XVIII, 220 p., Hardcover ISBN: 0-387-29146-6.