# Database Research at the National University of Singapore

Stephane Bressan Chee Yong Chan Wynne Hsu Mong-Li Lee
Tok-Wang Ling Beng Chin Ooi Kian-Lee Tan Anthony K.H. Tung
National University of Singapore, Singapore 117417

## 1. INTRODUCTION

At the National University of Singapore (NUS), the database group has worked on a wide range of research, ranging from traditional database technology (e.g., database design, query processing and optimization) to more advanced database technology (e.g., cloud and big data management) to novel database utilities (e.g., database usability, visualization, security and privacy). In this article, we describe some recent and on-going interdisciplinary projects for which we have received significant amount of funding.

## 2. CLOUD-BASED DATA MANAGEMENT

We have been developing efficient cloud computing platforms for large-scale services, and Big Data management and analytics using commodity hardware. We shall elaborate them below.

### 2.1 MapReduce-based Systems

One of our goals is to allow users of MapReduce-based systems to keep the programming model of the MapReduce framework, and yet to empower them with data management functionalities at an acceptable performance. We achieved this in two directions. First, we sought to identify key design factors of MapReduce (Hadoop) that affect its performance [17]. We conducted a comprehensive and in-depth study of Hadoop, and found that, by carefully tuning these factors, we can achieve much better performance. For example, MapReduce can benefit much from the use of indexes, and its performance can improve by a factor of 2.5 for selection tasks and a factor of up to 10 for join tasks. We also showed that, among the two types of I/O interfaces for scanning data, the direct I/O mode is superior over the streaming I/O mode.

Second, we have developed query processing engine under the MapReduce framework. At the operator level, we have developed join algorithms. In particular, our proposed MapReduce-based similarity (kNN) join exploits Voronoi diagram to minimize the number of objects to be sent to the `reducer` node to minimize computation and communication overheads [25]. We also designed several schemes for processing multi-join queries efficiently - while the Map-Join-Reduce mechanism [18] introduces a `join` operator to combine multiple datasets, the multi-join scheme in AQUA [40] exploits replication to expand the plan space. We have also developed an automatic query analyzer that accepts an SQL query, optimizes it and translates it into a set of MapReduce jobs [40]. Finally, to support data warehousing, we have leveraged on column store, and proposed Concurrent Join to support multi-way join over the partitioned data [20]. In all these works, we target to reduce the number of MapReduce jobs to minimize the initialization overheads.

### 2.2 epiC: A $V^3$-aware Data Intensive Cloud System

Our second direction is driven by the limitations of MapReduce-based systems to deal with "varieties" in the cloud data management. Most business production environments contain a mixture of data storage and processing systems; for example, customer data are maintained by a relational database and user requests are logged to a file system, while images and digital maps are handled by an object storage system. Processing and analyzing these data often requires different APIs and tools. SQL may be used for generating reports, while proprietary libraries may be used for feature extraction from images. Therefore, migrating such federated production systems into a centralized cloud infrastructure introduces three kinds of varieties (called $V^3$): variety of data (e.g., structured and unstructured), variety of storage (e.g., database and file systems), and variety of processing (e.g., SQL and proprietary APIs).

The $V^3$ problem mentioned above poses two main challenges to the cloud data management system: resource sharing and heterogeneous data processing. It is well known that deploying multiple storage systems on the same cloud can increase the utilization rate of underlying hardware since spaces released by one system can be reclaimed by another. However, the challenge is how to guarantee the performance isolation. For example, systems like HDFS or GFS are optimized for large sequential scanning and thus prefer manipulating large files. Sharing disks between such systems with the key-value stores may degrade their performance since key-value stores frequently create and delete small sized files, resulting in disk fragmentation.

MapReduce system is proven to be highly scalable for large scale data processing. But the system requires its users to re-implement their existing data processing algorithms with MapReduce interfaces. As an example, one must implement an SQL engine on top of MapReduce in order to perform SQL data processing. Such problem is not trivial for federated production systems, where multiple data formats have to be supported.

As a response to the $V^3$ challenge, we initiated the epiC project, a joint system project between researchers from NUS and Zhejiang University [2]. The goal of epiC is to provide a framework for facilitating companies to deploy and migrate their federated data systems to the cloud. The epiC system adopts an extensible design. The core of epiC provides two services: virtual block service (called VBS) which manages the cloud storage devices and a coordination framework (called $E^3$ [9]) which coordinates independent computations over federated systems. To analyze the data, users invoke a set of computing units (called Actors). In each Actor, users employ their favorite APIs to process a specific type of data and use $E^3$ to coordinate these Actors for producing the final results.

We have developed a novel elastic storage system ($ES^2$) [8] and deployed it on epiC. $ES^2$ employs vertical partitioning to group columns that are frequently accessed together, and horizontal partitioning to further split these column groups across a cluster of nodes. A number of novel cloud-based indexing structures (e.g., $B^+$-tree [39, 12], bitmap indexes [24], R-tree index [37]) have been developed.

We have also examined how transactions can be supported. This led to the design of ecStore [35]. ecStore exploits multi-version optimistic concurrency control and provides adaptive read consistency on replicated data.

### 2.3 Peer-to-Peer-based Cloud Data Management

Another direction that we are pursuing is the integration of cloud computing, database and peer-to-peer (P2P) technologies. Exploiting a P2P architecture on a cluster of nodes offers several advantages over the MapReduce framework: (a) It offers more robust query processing mechanisms as nodes can now communicate with one another; (b) It removes the single point-of-failure in the master/slave architecture of MapReduce; (c) It facilitates elastic design as peers can be readily added and removed in a P2P architecture.

**BestPeer++.** We have developed BestPeer++ [11, 10], a cloud-enabled evolution of BestPeer [26]. BestPeer++ is enhanced with distributed access control, multiple types of indexes, and pay-as-you-go query processing for delivering elastic data sharing services in the cloud. The software components of BestPeer++ are separated into two parts: *core* and *adapter*. The core contains all the data sharing functionalities and is designed to be platform independent. The adapter contains one *abstract* adapter which defines the elastic infrastructure service interface and a set of *concrete* adapter components which implement such an interface through APIs provided by specific cloud service providers (e.g., Amazon). We adopt this "two-level" design to achieve portability. BestPeer++ instances are organized as a structured P2P overlay network. We have used BATON [16], developed at NUS, as it can support range queries efficiently. The data are indexed by the table name, column name and data range for efficient retrieval.

**Katana.** The *Katana* framework is a novel peer-to-peer (P2P) based generalized data processing framework [14]. It can be deployed on many of the currently known structured P2P overlays. The framework provides a programming model in which processing logic may be implicitly distributed with universality and expressiveness, much like the MapReduce framework. The programming model can be distinguished into a data model and a processing model. We adopt a key-value data model with possible duplicated keys to represent the data elements. However, the data model is conceptually a graph-based model, i.e., data elements can be organized into a graph structure. Now, where the data is list-based, then the graph degenerates into a list. This facilitates the mapping from the data elements to the Cayley graphs which in turn can be mapped to the structured P2P overlays.

Like MapReduce, the Katana processing model hides the parallelism mechanism from the users. Instead, it provides two MapReduce-like functions:

`kata` and `ana`. However, unlike MapReduce, the `kata` and `ana` functions are independent from one another and are not required to be executed one after another. While `kata` jobs are used to perform aggregation of some sort over the data elements, `ana` jobs are used to build datasets based on the input data elements (i.e., to produce *data graphs* out of the input *data graph*). The execution essentially follows a post-order depth-first traversal of an arbitrary spanning tree of the data graph.

## 2.4 Big Data Projects

Our experience on managing data in the cloud has enabled us to participate in several large projects with substantial funding. The first, funded by the National Research Foundation of Singapore (NRF), focuses on exploiting cloud for large-scale data analytics in environmental monitoring and waste management in megacities [1]. This requires building a platform for scientists to manage and analyze large amount of sensor data collected from two cities (Singapore and Shanghai) in order to detect emergent pollutants and manage waste. Our initial effort is to develop LogBase, a scalable log-structured database system that adopts log-only storage to remove write bottleneck and to support fast system recovery [36]. In our current implementation, LogBase provides in-memory multi-version indexes and various primary and secondary log-based index to speed up retrieval of data from the log. In addition, LogBase supports transactions that bundle read and write operations spanning across multiple records.

The second project, also funded by NRF, aims to develop a comprehensive IT infrastructure for Big Data management, supporting data-intensive applications and analyses. Our epiC project has formed the basis for us to investigate various issues such as iterative computations that cannot be well supported by existing systems. At this moment, we are investigating check-pointing, recovery and concurrency issues in supporting iterative processing required for data analytics.

Finally, the third project comes under the Sensor-Enhanced Social Media (SeSaMe) Centre [3] jointly funded by Zhejiang University, NUS and Media Development Authority (MDA). The SeSaMe research center focuses on long-term research related to sensor-enhanced social media that enables linking of static and mobile cyber-physical environments over the Internet by the abstraction of sensing, processing, transport and presentation. The center will also facilitate the design of social media applications on cyber-physical systems through research advances that will transform the world by providing systems that respond more quickly. In this project, our goal is to leverage the Cloud techniques to efficiently manage and retrieve streaming data from sensors, mobile phones and other real-world data sources to support the analytical jobs of real world problem and a tool to visualize the results. We are building a new Cloud-based streaming engine to handle requests efficiently and reliably.

# 3. TSINGNUS: A LOCATION-BASED SERVICE SYSTEM TOWARDS LIVE CITY

The NUS-Tsinghua Extreme Search (NExT) Center [4], funded by the Media Development Authority (MDA) of Singapore, is a joint collaboration between the NUS and Tsinghua University to develop technologies towards a livable city. The program brings together researchers from different fields (multimedia, networks, databases) from the two universities to facilitate *extreme search* over large amount of real-time and dynamic data - social media (e.g, blogs, tweets, q&a forum), video, image, textual (documents) and structured data - beyond what is indexed in the web.

TsingNUS [6, 19] is a location-based service system that focuses on exploiting database technologies to support location-based services. TsingNUS goes beyond traditional location-aware applications that are based solely on user locations. Instead, TsingNUS aims to provide a more user-friendly location-aware search experience. First our *location-aware search-as-you-type* feature enables answers to be continuously returned and refined as users type in queries letter by letter [45]. For efficiency, we proposed the *prefix-region tree* (PR-tree), a tree-based index structure that organizes the dataspace into a hierarchy of spatial-textual regions such that (a) the spatial component of nodes nearer to the root are larger, and (b) the textual component of nodes nearer to the root are prefix of the textual component of descendant nodes.

Second, TsingNUS offers efficient mechanisms to process spatial-keyword queries for both AND semantics (where all keywords must appear in the retrieved content) and OR semantics (where some keywords appear in the retrieved content) [42]. Our newly developed scalable integrated inverted index, $I^3$, is an inverted index of *keyword cells*. A keyword cell denoted (keyword $w$, cell $c$) refers to a list of documents that contain $w$ and the associated spatial locality of the documents fall in region $c$. We have used the Quadtree structure to hierarchically partition the data space into cells.

Third, TsingNUS incorporates continuous spatial-keyword search to efficiently support continuously moving queries in a client-server system [15]. We have developed an effective model to represent the safe region of a moving top-$k$ spatial-keyword query. Such a region bounds the space for which the user (and hence the query) may move while the answers remain valid.

We are extending our work to road networks (e.g., finding frequent routes [7]) and to support a wider variety of query types (e.g., nearest group queries [41]). We are also exploring how users' social networks can be tapped upon to support more sophisticated queries.

# 4. INTEGRATED MINING AND VISUALIZATION OF COMPLEX DATA

The drive to find gold nuggets in data has resulted in the explosion of discovery algorithms in the past decade. Many of these discovery algorithms focus on specific data type. However, with the advances of technology, many applications now involve records with attributes of diverse data types, ranging from categorical, to numerical, to time series, to trajectories.

Knowing the relationships among all the different types of data can aid in the understanding of a patient health condition. For example, suppose we have a frequent itemset Male, Smoker and an interval-based temporal pattern Headache Overlap HighBloodPressure. If these two patterns occur together, it may raise an alarm as studies have shown that a male smoker who experiences headache with elevated blood pressure has a high risk of having cardiovascular disease.

Handling datasets with such variety is a challenge as the complexity of the problem can quickly grow out of hand. We have developed a framework to perform the integrated mining of big data with diverse data types [28]. The framework consists of algorithms for mining patterns from interval-based events [27], lag patterns involving motifs in time series data [29], spatial interaction patterns [32, 31], duration-aware region rules and path rules for trajectories [30]. With this, we are able to capture the associations among different complex data types and demonstrate how these patterns can be used to improve the classification accuracy in various real world datasets.

We have also developed a tool, in cooperation with the Center for Infectious Diseases Epidemiology and Research at the Saw Swee Hock School of Public Health, to generate and highlight interesting patterns discovered from the different data types. This tool will also allow the visualization of event incidences, clusters and heat maps. Ongoing research aims to develop an interactive system for the visualization and analysis of trajectories.

# 5. QUERY REVERSE ENGINEERING

To help users with constructing queries and understanding query results, we have developed an approach, termed Query by Output (QBO), to reverse engineer queries given an input pair of database and query output. Given a database D and a result table T = Q(D), which is the output of some query Q on D, the goal of QBO is to construct candidate queries Q', referred to as instance-equivalent queries, such that the output of query Q' on database D is equal to Q(D).

We have applied QBO to improve database usability in two contexts. In the first scenario, QBO is used to help users better understand their query results by augmenting the result of a query Q (w.r.t. a database) with instance-equivalent queries that describe alternative characterizations of their query results [34]. As an example, suppose that a university physician issues a query to his clinic's database to find students who have been infected with a skin rash over the past week. Besides returning the query result, if the database system had also computed and returned an equivalence-instance query that revealed the additional information that all the students in the query result either had recently returned from an overseas trip to region X or are staying in the same dormitory as those students, then the physician could have been alerted about a potential skin rash outbreak in those dormitories. Thus, it is useful to augment a query's result with alternative characterizations of the query's result to provide additional insightful information.

In the second scenario, QBO is used to generate explanations for unexpected query results that have missing expected result tuples [33]. As an example, suppose that a manager issues a query to compute the annual sales figures for each of her regional sales agents and she is surprised to find that Alice's sales performance is lower than that of Bob's, which is inconsistent with her impression of their results. The manager could issue a follow-up "why-not" question to clarify why Alice's sales figure is not higher than that of Bob's. Using QBO, the database system could respond to this why-not question with an explanation in the form of an alternative query (e.g., compute total sales for each sales agent excluding the period when Alice was on sick leave) which would have returned an output result that is consistent with the manager's why-not question. Thus, providing a capability to explain why-not questions would be very useful to help users understand their query results. We are currently implementating a query acquisition tool based on QBO that enables users to construct queries from examples of database and query result pairs.

## 6. DATA ANALYTICS

In addition to developing novel platforms for efficient data analytical processing, we are also looking at bringing human into the loop.

### 6.1 CrowdSourcing

We are developing a data analytics system that exploits crowdsourcing to manage complex tasks for which human can offer better (especially in terms of accuracy) alternative solutions. Our system, called Crowdsourcing Data Analytics System (CDAS), is designed to support deployment of crowdsourcing applications [23, 13]. In CDAS, a task is split into two parts - the computer-oriented tasks and human-oriented tasks. Crowdsourcing is employed to handle the human-oriented tasks. The results of the two tasks are then integrated. CDAS has a number of features that distinguish it from other crowdsourcing systems. First, CDAS has a quality-sensitive answering model that guides the crowdsourcing engine to process and monitor the human-oriented tasks. To reduce costs, the model employs a prediction model to estimate the number of workers required in order to achieve a certain level of accuracy. To ensure the quality of the estimation, historical information on reliability of workers is used. In fact, we also inject tasks for which answers are known in order to gauge the reliability of the workers. In addition, CDAS adopts a probabilistic approach (instead of the naive voting-based strategy) to verify the correctness of answers from workers. The idea of the scheme is to combine vote distribution of the current tasks and the historical accuracies and reliability of workers to determine the quality of the current answers by the workers. The intuition is to give higher weights to reliable workers.

Second, since workers complete their tasks asynchronously, CDAS supports "online aggregation", i.e., answers (with quality bounds) are continuously displayed and refined as responses from workers are received. This reduces the initial response time to end-users significantly.

We have demonstrated the effectiveness of CDAS in terms of both performance and ease of use in two different applications. A twitter sentiment analytics system has been developed on top of CDAS for analyzing the sentiments of movie goers. Another image tagging system has been built to facilitate image tagging of Flickr images. We have also exploited crowdsourcing in web table mapping and schema integration.

### 6.2 Collaborative Visual Analytics

In this research, we study how people can collaboratively achieve certain tasks by sharing their data and analytics results through the social network.

We have set up the *Internet Observatory* project [5] with the goals to monitor and analyze the dynamic user-generated contents on the Internet, and to provide a platform for users to share their findings. To provide context, we index these dynamic contents via Wikipedia, a well-established online encyclopedia which have entries for large number of entities and concepts [22, 21]. As an example, consider the Wikipedia entry for `Senkaku Island Dispute`. Besides visualizing the Wikepedia entry, our system also displays *dynamic* information (obtained from other sources) that are related to `Senkaku Island Dispute` including URLs, images, tag summarization, community view and geographical view. Currently, our system provides users with a set of social websites that they can choose to logon to in order to extract related information. This allows users to link/compare them to other information and opinions on the Internet. By doing so, the user is implicitly adding his/her private data into a public pool for general analysis.

We have also started the *ReadPeer* project which aims to promote reading as a large scale social activity by integrating ebooks and social networks to encourage more people to read and discuss about the materials they read. Our ReadPeer system allows users to make annotations on ebooks, research articles or any documents in PDF format. These annotations can be linked to various multimedia contents like blogs, videos, images, web links etc. and shared to friends in a social network.

Our approach to collaborative visual analytics involve reorganizing social media messages around a center of focus like Wikipedia articles or ebooks instead of putting these messages in a plain news feed. This allows users of common interest to come together to share their insights and analysis. Central to this is the design of visual interfaces that allow users to communicate and understand each other's perspectives. Moreover, these interactions generate databases that capture a lot of interesting semantics through linkages of social media messages into a rich information network. Visualizing such a rich information network is challenging [43, 44, 38].

## 7. ACKNOWLEDGEMENTS

Many of our research are done in collaboration with international visitors. These include Divy Agrawal, Elisa Bertino, H.V. Jagadish, David Maier and Tamer Ozsu. We also thank our research fellows for their contributions to our group. Finally, special thanks to our many graduate students - without them, we would not be where we are today!

## 8. REFERENCES


[1] Energy and environmental sustainability solutions for megacities. http://www.nus.edu.sg/neri/E2S2.html, 2013.

[2] epiC@NUS. http://www.comp.nus.edu.sg/~epic, 2013.

[3] SeSaMe. http://sesame.comp.nus.edu.sg/, 2013.

[4] The NExT Center. http://next.comp.nus.edu.sg, 2013.

[5] Trendspedia. http://www.trendspedia.com/, 2013.

[6] TsingNUS. http://tsingnus.comp.nus.edu.sg, 2013.

[7] H. Aung, L. Guo, and K. L. Tan. Mining sub-trajectory cliques to find frequent routes. In *SSTD*, 2013.

[8] Y. Cao, C. Chen, F. Guo, D. Jiang, Y. Lin, B. C. Ooi, H. T. Vo, S. Wu, and Q. Xu. $\mathrm{Es}^2$: A cloud data storage system for supporting both oltp and olap. In *ICDE*, pages 291–302, 2011.

[9] G. Chen, K. Chen, D. Jiang, B. C. Ooi, L. Shi, H. T. Vo, and S. Wu. E3: an elastic execution engine for scalable data processing. *JIP*, 20(1):65–76, 2012.

[10] G. Chen, T. Hu, D. Jiang, P. Lu, K. L. Tan, H. T. Vo, and S. Wu. Bestpeer++: A peer-to-peer based large-scale data processing platform. In *TKDE (Special Issue for Best Papers in ICDE'2012)*.

[11] G. Chen, T. Hu, D. Jiang, P. Lu, K. L. Tan, H. T. Vo, and S. Wu. Bestpeer++: A peer-to-peer based large-scale data processing platform. In *ICDE*, pages 582–593, 2012.

[12] G. Chen, H. T. Vo, S. Wu, B. C. Ooi, and M. T. Özsu. A framework for supporting dbms-like indexes in the cloud. *PVLDB*, 4(11):702–713, 2011.

[13] J. Gao, X. Liu, B. C. Ooi, H. Wang, and G. Chen. An online cost sensitive decision-making method in crowdsourcing systems. In *SIGMOD Conference*, 2013.

[14] W. X. Goh and K. L. Tan. Katana: Generalized data processing on peer-to-peer overlays. In *IC2E*, 2013.

[15] W. Huang, G. Li, K. L. Tan, and J. Feng. Efficient safe-region construction for moving top-k spatial keyword queries. In *CIKM*, pages 932–941, 2012.

[16] H. V. Jagadish, B. C. Ooi, and Q. H. Vu. Baton: A balanced tree structure for peer-to-peer networks. In *VLDB*, pages 661–672, 2005.

[17] D. Jiang, B. C. Ooi, L. Shi, and S. Wu. The performance of mapreduce: An in-depth study. *PVLDB*, 3(1):472–483, 2010.

[18] D. Jiang, A. K. H. Tung, and G. Chen. Map-join-reduce: Toward scalable and efficient data analysis on large clusters. *IEEE Trans. Knowl. Data Eng.*, 23(9):1299–1311, 2011.

[19] G. Li, N. Zhang, R. Zhong, W. Huang, K. L. Tan, J. Feng, and L. Zhou. TsingNUS: A location-based service system towards live city (demo). In *SIGMOD*, 2013.

[20] Y. Lin, D. Agrawal, C. Chen, B. C. Ooi, and S. Wu. Llama: leveraging columnar storage for scalable join processing in the mapreduce framework. In *SIGMOD Conference*, pages 961–972, 2011.

[21] C. Liu, B. Cui, and A. K. H. Tung. Integrating web 2.0 resources by wikipedia. In *ACM Multimedia*, pages 707–710, 2010.

[22] C. Liu, S. Wu, S. Jiang, and A. K. H. Tung. Cross domain search by exploiting wikipedia. In *ICDE*, pages 546–557, 2012.

[23] X. Liu, M. Lu, B. C. Ooi, Y. Shen, S. Wu, and M. Zhang. Cdas: A crowdsourcing data analytics system. *PVLDB*, 5(10):1040–1051, 2012.

[24] P. Lu, S. Wu, L. Shou, and K. L. Tan. An efficient and compact indexing scheme for large-scale data store. In *ICDE*, 2013.

[25] W. Lu, Y. Shen, S. Chen, and B. C. Ooi. Efficient processing of k nearest neighbor joins using mapreduce. *PVLDB*, 5(10):1016–1027, 2012.

[26] W. S. Ng, B. C. Ooi, K. L. Tan, and A. Zhou. Peerdb: A p2p-based system for distributed data sharing. In *ICDE*, pages 633–644, 2003.

[27] D. Patel, W. Hsu, and M. L. Lee. Mining relationships among interval-based events for classification. In *SIGMOD Conference*, pages 393–404, 2008.

[28] D. Patel, W. Hsu, and M. L. Lee. Integrating frequent pattern mining from multiple data domains for classification. In *ICDE*, pages 1001–1012, 2012.

[29] D. Patel, W. Hsu, M. L. Lee, and S. Parthasarathy. Lag patterns in time series databases. In *DEXA (2)*, pages 209–224, 2010.

[30] D. Patel, C. Sheng, W. Hsu, and M. L. Lee. Incorporating duration information for trajectory classification. In *ICDE*, pages 1132–1143, 2012.

[31] C. Sheng, W. Hsu, M. L. Lee, and A. K. H. Tung. Discovering spatial interaction patterns. In *DASFAA*, pages 95–109, 2008.

[32] C. Sheng, Y. Zheng, W. Hsu, M. L. Lee, and X. Xie. Answering top-$k$ similar region queries. In *DASFAA (1)*, pages 186–201, 2010.

[33] Q. T. Tran and C. Y. Chan. How to conquer why-not questions. In *SIGMOD Conference*, pages 15–26, 2010.

[34] Q. T. Tran, C. Y. Chan, and S. Parthasarathy. Query by output. In *SIGMOD Conference*, pages 535–548, 2009.

[35] H. T. Vo, C. Chen, and B. C. Ooi. Towards elastic transactional cloud storage with range query support. *PVLDB*, 3(1):506–517, 2010.

[36] H. T. Vo, S. Wang, D. Agrawal, G. Chen, and B. C. Ooi. Logbase: A scalable log-structured database system in the cloud. *PVLDB*, 5(10):1004–1015, 2012.

[37] J. Wang, S. Wu, H. Gao, J. Li, and B. C. Ooi. Indexing multi-dimensional data in a cloud system. In *SIGMOD Conference*, pages 591–602, 2010.

[38] N. Wang, S. Parthasarathy, K. Tan, and A. K. H. Tung. Csv: visualizing and mining cohesive subgraphs. In *SIGMOD Conference*, pages 445–458, 2008.

[39] S. Wu, D. Jiang, B. C. Ooi, and K. L. Wu. Efficient b-tree based indexing for cloud data processing. *PVLDB*, 3(1):1207–1218, 2010.

[40] S. Wu, F. Li, S. Mehrotra, and B. C. Ooi. Query optimization for massively parallel data processing. In *ACM SOCC*, 2011.

[41] D. Zhang, C. Y. Chan, and K. L. Tan. Nearest group queries. In *SSDBM*, 2013.

[42] D. Zhang, K. L. Tan, and A. K. H. Tung. Scalable top-k spatial keyword search. In *EDBT*, pages 359–370, 2013.

[43] F. Zhao, G. Das, K. Tan, and A. K. H. Tung. Call to order: a hierarchical browsing approach to eliciting users' preference. In *SIGMOD Conference*, pages 27–38, 2010.

[44] F. Zhao and A. K. H. Tung. Large scale cohesive subgraphs discovery for social network visual analysis. *PVLDB*, 6(2), 2012.

[45] R. Zhong, J. Fan, G. Li, K. L. Tan, and L. Zhou. Location-aware instant search. In *CIKM*, 2012.