The Era of Big Spatial Data: A Survey

Ahmed Eldawy
University of California, Riverside
eldawy@cs.ucr.edu

Mohamed F. Mokbel
University of Minnesota
mokbel@cs.umn.edu

the essence of knowledge
Boston — Delft

Foundations and Trends® in Databases

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

A. Eldawy and M. F. Mokbel. *The Era of Big Spatial Data: A Survey*. Foundations and Trends® in Databases, vol. 6, no. 3-4, pp. 163–273, 2013.

This Foundations and Trends® issue was typeset in LaTeX using a class file designed by Neal Parikh. Printed on acid-free paper.

ISBN: 978-1-68083-225-9

Editorial Scope

Topics

Foundations and Trends® in Databases covers a breadth of topics relating to the management of large volumes of data. The journal targets the full scope of issues in data management, from theoretical foundations, to languages and modeling, to algorithms, system architecture, and applications. The list of topics below illustrates some of the intended coverage, though it is by no means exhaustive:

- Data models and query languages
- Query processing and optimization
- Storage, access methods, and indexing
- Transaction management, concurrency control, and recovery
- Deductive databases
- Parallel and distributed database systems
- Database design and tuning
- Metadata management
- Object management
- Trigger processing and active databases
- Data mining and OLAP
- Approximate and interactive query processing
- Data warehousing
- Adaptive query processing
- Data stream management
- Search and query integration
- XML and semi-structured data
- Web services and middleware
- Data integration and exchange
- Private and secure data management
- Peer-to-peer, sensornet, and mobile data management
- Scientific and spatial data management
- Data brokering and publish/subscribe
- Data cleaning and information extraction
- Probabilistic data management

Information for Librarians

Foundations and Trends® in Databases, 2013, Volume 6, 4 issues. ISSN paper version 1931-7883. ISSN online version 1931-7891. Also available as a combined paper and online subscription.

Foundations and Trends® in Databases
Vol. 6, No. 3-4 (2013) 163–273

DOI: 10.1561/1900000054

The Era of Big Spatial Data: A Survey

Ahmed Eldawy
University of California, Riverside
eldawy@cs.ucr.edu

Mohamed F. Mokbel
University of Minnesota
mokbel@cs.umn.edu

Contents

Abstract

The recent explosion in the amount of spatial data calls for specialized systems to handle big spatial data. In this survey, we summarize the state-of-the-art work in the area of big spatial data. We categorize the existing work in this area according to six different angles, namely, *approach*, *architecture*, *language*, *indexing*, *querying*, and *visualization*. (1) The approaches used to implement spatial query processing can be categorized as *on-top*, *from-scratch* and *built-in* approaches. (2) The existing works follow different *architectures* based on the underlying system they extend such as MapReduce, key-value stores, or parallel DBMS. (3) The high-level *language* of the system is the main interface that hides the complexity of the system and makes it usable for non-technical users. (4) The spatial *indexing* is the key feature of many systems which allows them to achieve orders of magnitude performance speedup by carefully laying out data in the distributed storage. (5) The *query processing* is at the heart of all the surveyed systems as it defines the types of queries supported by the system and how efficiently they are implemented. (6) The *visualization* of big spatial data is how the system is capable of generating images that describe terabytes of data to help users explore them. This survey describes each of these components, in detail, and gives examples of how they are implemented in existing systems. At the end, we give case studies of real applications that make use of these systems to provide services for end users.

A. Eldawy and M. F. Mokbel. *The Era of Big Spatial Data: A Survey*. Foundations and Trends® in Databases, vol. 6, no. 3-4, pp. 163–273, 2013.

DOI: 10.1561/1900000054.

1

Introduction

There has been a recent marked increase in the amount of spatial data collected by new devices such as smart phones, space telescopes, and medical devices, among others. For example, space telescopes generate up to 150 GB weekly of spatial data [67], medical devices produce spatial images (X-rays) at a rate of 50 PB per year [123], a NASA archive of satellite earth images has more than 1 PB and increases daily by 200 GB [75], while there are 10 Million geotagged tweets created in Twitter every day as 2% of the whole Twitter firehose [58, 114]. Meanwhile, various applications and agencies need to process an unprecedented amount of spatial data. For example, the Blue Brain Project [86, 111] studies the brain's architectural and functional principles through modeling brain neurons as spatial data. Epidemiologists use spatial analysis techniques to identify cancer clusters [97], track infectious disease [15], and follow drug addiction [112]. Meteorologists study and simulate climate data through spatial analysis [50, 51, 52]. News reporters use geotagged tweets for event detection and analysis [100].

Due to this rise in the volume of spatial data, it becomes highly desirable for researchers and developers to be able to process them using big data frameworks, such as MapReduce [31], Hadoop [63], Hive [113], BigTable [29], HBase [65], Impala [70, 118], Dremel [88, 87], Vertica [108], Dryad [68], AsterixDB [10], and Spark [129]. While these systems are general purpose and can process spatial data, they provide sub-par performance due to the lack of specialized components that are designed for spatial data. In other words, they deal with spatial data in the same way they do with any other data while largely ignoring the inherent properties of spatial data and spatial query processing. For example, it was shown in different systems that the use of spatial indexes can provide orders of magnitude speedup to simple queries such as range query and k nearest neighbor (kNN) [44, 122, 79, 124, 126].

To fill in the gap between spatial data processing and big data frameworks, several research attempts have been made to extend these frameworks to better handle and process spatial data, such as Hadoop-GIS [4], SpatialHadoop [41, 44, 38, 43], $\mathcal{MD}$-HBase [92, 91], Parallel Secondo [76, 77, 78, 61, 90], and ESRI Tools for Hadoop [122]. In this survey, we summarize the state-of-the-art techniques in processing Big Spatial Data while highlighting open research problems and identifying research trends. This survey aims to be very helpful for existing researchers and developers working in the area of Big Spatial Data to understand the existing work, as well as for future researchers who are interested in pursuing research in this area.

Big data is usually characterized by its Volume, Velocity, Variety, and Veracity, which all apply to spatial data. The *volume* of big data is increasing tremendously due to the automated and continuous acquisition of data and the high resolution of such data. For example, the size of the LP DAAC archive [75] exceeded one petabyte with a highest resolution of 250 meters while the European Space Agency (ESA) has recently released data from the Sentinel-2 mission with up-to 1 meter resolution data [102]. The *velocity* of spatial data has also increased with the ubiquity of small devices capable of generating small amounts of data at excessively high rates, such as GPS tracking, Facebook comments, and POS transactions. These sources can generate at least 1 petabyte per year [85]. The big *variety* of spatial data emerges from the different data types, e.g., point, line, polygon, and raster images; different data formats, e.g., Shapefile, GeoJSON, and KML; and various projections in the case of geographical data, e.g., WGS84 and Sinusoidal. Combining these different data sources together imposes huge challenges to applications that deal with big spatial data. Finally, there are different sources of low *veracity* associated with spatial data including the inherent errors in localization techniques, e.g., GPS, WiFi, and cellular triangulation, and the noise in the data collected by satellites due to clouds and mis-alignment of satellite images.

This survey classifies existing work by considering six aspects of big spatial data systems: (1) The implementation *approach*, which defines whether it is implemented *on-top* of an existing system, *built inside* its core, or developed completely *from scratch*. (2) The underlying *architecture*, which describes the primary processing model of the systems, such as parallel DBMS, Message Passing Interface (MPI), MapReduce, key-value store, array DB, Resilient Distributed Datasets (RDD), or Hyracks. (3) The high-level *language* of the system, if any exists. (4) The existence of *spatial indexes* in the system and the types of these indexes. (5) The types of *queries* supported by the system, such as range query, spatial join, computational geometry, or spatial data mining. (6) The support of big spatial data *visualization* in the system.

Table 1.1 outlines the surveyed work in the area of big spatial data. Each row represents a system or a body of work related to big spatial data, while each column represents one of the six aspects that we will discuss, namely, *approach*, *architecture*, *language*, *indexing*, *querying*, and *visualization*. The following chapters in the survey will delve into the details of each of these aspects (i.e., the table's columns) to provide more details about them.

Implementation Approach: As shown in the second column of Table 1.1, the surveyed work can be categorized according to the implementation approach into three main categories, *on-top*, *from-scratch*, and *built-in*. In the *on-top* approach, an existing system is used as a black box while the logic of spatial data is provided as user-defined functions (UDFs). While this approach is simple to implement and portable to different releases of the underlying system, it usually suf-

	Approach	Architecture	Language	Indexes	Queries	Visualization
Paradise [96, 34, 127]	From-scratch	Parallel DB	SQL	Grid	RQ, SJ, Raster	Single level
Parallel Secondo [76]	Built-in	Parallel DB	SQL-Like	Local only	RQ, SJ	-
Sphinx [39]	Built-in	Parallel DB	SQL	R-tree, Quad tree	RQ, SJ	-
SciDB [110, 98, 17]	From-scratch	Array DB	AQL, AFL	Kd tree	RQ, KNN	Single/Multi
RasDaMan [18, 20, 21, 19]	From-scratch	Array DB	RasQL	-	Raster	Single level
$\mathcal{MD}$-HBase [92]	Built-in	KV store	-	Quad Tree, Kd tree	RQ, KNN	-
GeoMesa [54]	Built-in	KV store	CQL*	Geohash	RQ	Via GeoServer
EMINC [133]	From-scratch	MPI	-	Kd tree, R-tree	RQ, K-means, DBSCAN	-
R-tree construction [27]	On-top	MapReduce	-	R-tree	Image quality	-
SJMR [131, 132, 121, 73]	On-top	MapReduce	-	R-tree	RQ, KNN, SJ, ANN	-
K-Means [134]	On-top	MapReduce	-	-	K-means	-
MR-DBSCAN [66]	On-top	MapReduce	-	-	DBSCAN	-
Voronoi Diagram [5]	On-top	MapReduce	-	-	VD, NN Queries	-
3D Visualization [117]	On-top	MapReduce	-	-	-	Single level
KNN Join [80, 130]	On-top	MapReduce	-	-	KNN Join	-
Multiway SJ [60]	On-top	MapReduce	-	-	Multiway SJ	-
BRACE [120]	From-scratch	MapReduce	BRASIL	Grid	SJ	-
PRADASE [81]	Built-in	MapReduce	-	Quad-tree	RQ	
Hadoop GIS [4]	Built-in	MapReduce	QL^{SP}	Grid	RQ, KNN, SJ	-
SpatialHadoop [44, 40, 46]	Built-in	MapReduce	Pigeon*	R tree/Quad tree	RQ, KNN, SJ, CG	Single/Multi
ScalaGiST [79]	Built-in	MapReduce	-	GiST	RQ, KNN	-
Esri Tools [122]	Built-in	MapReduce	HiveQL*	PMR Quad Tree	RQ, KNN	-
ISP-MC[125]	On-top	RDD	Scala-based	On-the-fly	SJ	-
GeoTrellis [69]	On-top	RDD	Scala-based	-	Map Algebra	-
GeoSpark [126]	Built-in	RDD	Scala-based	R-tree, Quad-tree	RQ, KNN, SJ	-
Simba [124]	Built-in	RDD	SQL	R-tree	RQ, KNN, SJ, KNN-Join	-
Asterix-DB [10, 11, 7]	Built-in	Hyracks	AQL	R-tree local index	RQ	-

∗ OGC-compliant

Table 1.1: Existing work in the area of big spatial data

fers from poor performance due to the underlying system being unaware of the properties of spatial data. The *from-scratch* approach is the other extreme, where a new system is built from scratch to support big spatial data processing. This allows the system to achieve very high performance on spatial queries as the core is customized for this kind of data. However, it becomes very hard to maintain and might be impractical if users wish to mix spatial with non-spatial query processing. The *built-in* approach balances efficiency with simplicity as it injects spatial data awareness inside an existing general-purpose system. This makes it efficient since the internal system becomes aware of spatial data and still it is not as complicated as building an entire system from scratch. Besides, it is more practical for users who wish to mix spatial and non-spatial workloads as it maintains the efficiency of the system with non-spatial data. The main drawback is that if the spatial extension is built on a side branch of the general-purpose system's code base, the built-in system then becomes tied to a specific version of the underlying general-purpose system and cannot be easily ported to newer versions. The three approaches are further described in Chapter 2.

Architecture: The systems that are discussed in this survey typically follow one of the standard approaches used in other big data systems, such as parallel DBMS, key-value stores, array databases, message passing interface (MPI), MapReduce, resilient distributed datasets (RDD), or Hyracks, as described in the third column of Table 1.1. Some of these surveyed systems might modify the underlying system to better support spatial data but they still preserve its architecture. The choice of a specific architecture to use depends mainly on the type of application that needs to be supported and the types of queries that will run on it. For example, MapReduce is designed for lengthy analytic queries that need to spill most of their intermediate data to disk, while RDD is more geared towards iterative jobs that can afford storing all of their data in main memory. The different architectures are described in detail in Chapter 3.

Language: The fourth column of Table 1.1 shows examples of high level languages supported in big spatial data systems. A high level language is extremely important, as it allows non-technical users to easily interact with the system. There are some industry standards for spatial data types and operations that are supported by existing systems for spatial data including PostGIS [99], Oracle Spatial [71], and ESRI ArcGIS [12]. It is highly desirable for big spatial data systems to support these standards to make it easier to adopt for users who are already familiar with them. The details of the high level languages are given in Chapter 4.

Indexes: Spatial indexes define an efficient way for storing data such that some queries run more efficiently. The fifth column of Table 1.1 shows the different types of indexes supported in the surveyed work. While there are many in-memory and on-disk index structures used in traditional systems, they cannot be used as-is in distributed systems due to the different storage and processing models used in such systems. Most distributed systems follow a two-layer index design of one *global index*, which partitions data across machines, and multiple *local indexes*, which organize records inside each machine. By controlling how the global and local indexes are constructed, a wide range of spatial indexes can be realized for big spatial data. Spatial indexes are further described in Chapter 5.

Queries: The main functionality of big spatial data systems is query processing, which performs spatial operations on the data. As shown in the sixth column of Table 1.1, we categorize queries into five categories as follows: (1) Basic queries such as point queries, range queries, and nearest neighbor queries. (2) Spatial join queries such as self-join, binary join, multi-way join, and kNN join. (3) Computational geometry queries such as polygon union, convex hull, skyline, and Voronoi diagram construction. (4) Spatial data mining such as the k-means and DBSCAN clustering algorithms. (5) Raster operations that deal with raster data represented as two-dimensional arrays of values. More details about query processing are given in Chapter 6.

Visualization: A highly desirable feature of data management in general, and for spatial data in particular, is visualization, which is the process of generating an image that describes an underlying dataset.

Visualization is an international communication language which allows users to spot interesting patterns that are very hard to notice otherwise. Some systems support only single-level image visualization that produces a single image with a fixed resolution, while other systems provide multi-level image visualization with the ability to interactively zoom in or out to see more or less details. The seventh column of Table 1.1 shows the types of visualization supported in each of the surveyed systems. Visualization is further explained in Chapter 7.

Datasets: To help system and application developers, this survey also provides references to several big spatial datasets that are publicly available and can be used for benchmarking or testing the systems. These datasets cover different types of data sources that can serve a wide range of applications, such as rich maps for the whole world, real trips made by taxi cabs in New York City, world-wide geo-tagged events collected since 1979, and a 1 PB archive of daily satellite data for the whole world over a period of 15 years. Details of the datasets will be provided in Chapter 8.

Applications: To make it easier to comprehend the whole survey, we provide several case studies of end-user applications that process big spatial data. These include SHAHED [37, 45], a system for analyzing satellite data using MapReduce, EarthDB [98], which uses SciDB for processing satellite data, TAREEG [9, 8], a web-based system for extracting world-wide map information, Taghreed [83, 84], which analyzes and visualizes geotagged tweets, AscotDB [115], a system for querying and analyzing astronomical data using SciDB, and GISQF [6], a MapReduce-based system for processing world-wide geotagged events.

It is important to mention that the above dimensions are not completely independent and they are usually application-driven. For example, an application for analyzing historical data might prefer the MapReduce architecture and support analytical queries, such as spatial join or kNN join, while spatial indexes might be of less importance. On the other hand, an application for exploring streaming data, e.g., geotagged tweets, would benefit from key-value stores that support fast rates of insertion and deletion, with spatial indexes being an important part of the system to efficiently answer interactive queries such as point and range selections. In this survey, we provide a few examples of applications that will help readers understand how these dimensions are related.

The rest of this survey is organized as follows. Chapter 2 describes the different implementation approaches. Chapter 3 discusses the various underlying architectures. Chapter 4 lays out the current work in spatial languages for big spatial data systems. Chapter 5 provides the details of big spatial data indexes. Chapter 6 describes the details of query processing on big spatial data. Chapter 7 discusses recent work in the area of big spatial data visualization. Chapter 8 provides some references to real big spatial datasets. Finally, Chapter 9 concludes the paper with several case studies of applications for big spatial data.

References

[1] Accumulo. Available at `http://accumulo.apache.org`, 2015.

[2] Ablimit Aji, Xiling Sun, Hoang Vo, Qiaoling Liu, Rubao Lee, Xiaodong Zhang, Joel H. Saltz, and Fusheng Wang. Demonstration of Hadoop-GIS: a spatial data warehousing system over MapReduce. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 518–521, Orlando, FL, November 2013.

[3] Ablimit Aji, George Teodoro, and Fusheng Wang. Haggis: Turbocharge a MapReduce Based Spatial Data Warehousing System with GPU Engine. In *Proceedings of the ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data, BigSpatial*, pages 15–20, Dallas, TX, November 2014.

[4] Ablimit Aji, Fusheng Wang, Hoang Vo, Rubao Lee, Qiaoling Liu, Xiaodong Zhang, and Joel H. Saltz. Hadoop-GIS: A High Performance Spatial Data Warehousing System over MapReduce. *Proceedings of the VLDB Endowment, PVLDB*, 6(11):1009–1020, 2013.

[5] Afsin Akdogan, Ugur Demiryurek, Farmoush Banaei-Kashani, and Cyrus Shahabi. Voronoi-based Geospatial Query Processing with MapReduce. In *International Conference on Cloud Computing Technology and Science, CloudCom*, pages 9–16, Indianapolis, IN, November 2010.

[6] Khaled Mohammed Al-Naami, Sadi Evren Seker, and Latifur Khan. GISQF: An Efficient Spatial Query Processing System. In *International Conference on Cloud Computing Technology and Science, CloudCom*, pages 681–688, Anchorage, AK, June 2014.

[7] Abdullah A. Alamoudi, Raman Grover, Michael J. Carey, and Vinayak R. Borkar. External Data Access And Indexing In AsterixDB. In *Proceedings of the International Conference on Information and Knowledge Management, CIKM*, pages 3–12, Melbourne, VIC, Australia, October 2015.

[8] Louai Alarabi, Ahmed Eldawy, Rami Alghamdi, and Mohamed F. Mokbel. TAREEG: A MapReduce-Based System for Extracting Spatial Data from OpenStreetMap. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 83–92, Dallas, TX, November 2014.

[9] Louai Alarabi, Ahmed Eldawy, Rami Alghamdi, and Mohamed F. Mokbel. TAREEG: a MapReduce-based Web Service for Extracting Spatial Data from OpenStreetMap. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 897–900, Snowbird, UT, June 2014.

[10] Sattam Alsubaiee, Yasser Altowim, Hotham Altwaijry, Alexander Behm, Vinayak R. Borkar, Yingyi Bu, Michael J. Carey, Inci Cetindil, Madhusudan Cheelangi, Khurram Faraaz, Eugenia Gabrielova, Raman Grover, Zachary Heilbron, Young-Seok Kim, Chen Li, Guangqiang Li, Ji Mahn Ok, Nicola Onose, Pouria Pirzadeh, Vassilis J. Tsotras, Rares Vernica, Jian Wen, and Till Westmann. AsterixDB: A Scalable, Open Source BDMS. *Proceedings of the VLDB Endowment, PVLDB*, 7(14):1905–1916, 2014.

[11] Sattam Alsubaiee, Alexander Behm, Vinayak Borkar, Zachary Heilbron, Young-Seok Kim, Michael J. Carey, Markus Dreseler, and Chen Li. Storage Management in AsterixDB. *Proceedings of the VLDB Endowment, PVLDB*, 7(10):841–852, June 2014.

[12] Esri ArcGIS. Available at `http://www.esri.com/software/arcgis/`, 2015.

[13] Michael Armbrust, Reynold S. Xin, Cheng Lian, Yin Huai, Davies Liu, Joseph K. Bradley, Xiangrui Meng, Tomer Kaftan, Michael J. Franklin, Ali Ghodsi, and Matei Zaharia. Spark SQL: Relational Data Processing in Spark. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 1383–1394, Melbourne, Victoria, Australia, May 2015.

[14] Apache AsterixDB. Available at `http://asterixdb.apache.org`, 2015.

[15] A. Auchincloss, S. Gebreab, C. Mair, and A. Diez Roux. A Review of Spatial Methods in Epidemiology: 2000-2010. *Annual Review of Public Health*, 33:107–22, April 2012.

[16] Shivnath Babu and Herodotos Herodotou. Massively Parallel Databases and MapReduce Systems. *Foundations and Trends® in Databases*, 5(1):1–104, 2013.

[17] Leilani Battle, Remco Chang, and Michael Stonebraker. Dynamic Prefetching of Data Tiles for Interactive Visualization. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 1363–1375, San Francisco, CA, June 2016.

[18] Peter Baumann, Andreas Dehmel, Paula Furtado, Roland Ritsch, and Norbert Widmann. The multidimensional database system rasdaman. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 575–577, Seattle, WA, June 1998.

[19] Peter Baumann, Andreas Dehmel, Paula Furtado, Roland Ritsch, and Norbert Widmann. Spatio-Temporal Retrieval with RasDaMan. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 746–749, Edinburgh, Scotland, UK, September 1999.

[20] Peter Baumann, Paula Furtado, Roland Ritsch, and Norbert Widmann. Geo/Environmental and Medical Data Management in the RasDaMan System. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 548–552, Athens, Greece, August 1997.

[21] Peter Baumann, Paula Furtado, Roland Ritsch, and Norbert Widmann. The RasDaMan Approach to Multidimensional Database Management. In *Proceedings of the ACM Symposium on Applied Computing, SAC*, pages 166–173, Sane Jose, CA, March 1997.

[22] Bill Beauregard, Souri Das, Matt Perry, and Zhe Wu. Oracle Spatial and Graph: Benchmarking a Trillion Edges RDF Graph. Technical report, Oracle, November 2016.

[23] Jon Louis Bentley. Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM*, 18(9):509–517, 1975.

[24] Mark De Berg, Otfried Cheong, Marc Van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer, 2008.

[25] Marsha J. Berger and Shahid H. Bokhari. A Partitioning Strategy for Nonuniform Problems on Multiprocessors. *IEEE Transactions on Computers*, C-36(5):570–580, May 1987.

[26] Michael J. Carey, David J. DeWitt, Daniel Frank, M. Muralikrishna, Goetz Graefe, Joel E. Richardson, and Eugene J. Shekita. The Architecture of the EXODUS Extensible DBMS. In *Proceedings on the International Workshop on Object-oriented Database Systems*, pages 52–65, Pacific Grove, CA, 1986.

[27] Ariel Cary, Zhengguo Sun, Vagelis Hristidis, and Naphtali Rishe. Experiences on Processing Spatial Data with MapReduce. In *Proceedings of the International Conference on Scientific and Statistical Database Management, SSDBM*, pages 302–319, New Orleans, Louisiana, June 2009.

[28] Ronnie Chaiken, Bob Jenkins, Per-Åke Larson, Bill Ramsey, Darren Shakib, Simon Weaver, and Jingren Zhou. SCOPE: Easy and Efficient Parallel Processing of Massive Data Sets. *Proceedings of the VLDB Endowment, PVLDB*, 1(2):1265–1276, 2008.

[29] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Michael Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber. Bigtable: A Distributed Storage System for Structured Data. *ACM Transactions Computer Systems, TOCS*, 26(2), 2008.

[30] Irving Cordova and Teng-Sheng Moh. DBSCAN on Resilient Distributed Datasets. In *International Conference on High Performance Computing Simulation, HPCS*, pages 531–540, Amsterdam, the Netherlands, July 2015.

[31] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, 51:107–113, 2008.

[32] Jochen Van den Bercken, Bernhard Seeger, and Peter Widmayer. The Bulk Index Join: A Generic Approach to Processing Non-Equijoins. In *Proceedings of the International Conference on Data Engineering, ICDE*, page 257, Sydney, Australia, March 1999.

[33] David J. DeWitt, Shahram Ghandeharizadeh, Donovan A. Schneider, Allan Bricker, Hui-I Hsiao, and Rick Rasmussen. The Gamma Database Machine Project. *IEEE Transactions on Knowledge and Data Engineering, TKDE*, 2(1):44–62, 1990.

[34] David J. DeWitt, Navin Kabra, Jun Luo, Jignesh M. Patel, and Jie-Bing Yu. Client-Server Paradise. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 558–569, Santiago de Chile, Chile, September 1994.

[35] Jens-Peter Dittrich and Bernhard Seeger. Data Redundancy and Duplicate Detection in Spatial Join Processing. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 535–546, San Diego, CA, February 2000.

[36] Ahmed Eldawy. SpatialHadoop: Towards Flexible and Scalable Spatial Processing using MapReduce. In *The PhD Symposium in the International Conference on Management of Data, SIGMOD*, pages 46–50, Snowbird, UT, June 2014.

[37] Ahmed Eldawy, Saif Al-Harthi, Abdulhadi Alzaidy, Anas Daghistani, Sohaib Ghani, Saleh Basalamah, and Mohamed F. Mokbel. A Demonstration of Shahed: A MapReduce-based System for Querying and Visualizing Satellite Data. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1444–1447, Seoul, Korea, April 2015.

[38] Ahmed Eldawy, Louai Alarabi, and Mohamed F. Mokbel. Spatial Partitioning Techniques in SpatialHadoop. In *Proceedings of the VLDB Endowment, PVLDB*, pages 1602–1605, Kohala Coast, HI, September 2015.

[39] Ahmed Eldawy, Mostafa Elganainy, Ammar Bakeer, Ahmed Abdelmotaleb, and Mohamed F. Mokbel. Sphinx: Distributed Execution of Interactive SQL Queries on Big Spatial Data. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 78:1–78:4, Seattle, WA, November 2015.

[40] Ahmed Eldawy, Yuan Li, Mohamed F. Mokbel, and Ravi Janardan. CG_Hadoop: Computational Geometry in MapReduce. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 284–293, Orlando, FL, November 2013.

[41] Ahmed Eldawy and Mohamed F. Mokbel. A demonstration of spatialhadoop: An efficient mapreduce framework for spatial data. *Proceedings of the VLDB Endowment, PVLDB*, 6(12):1230–1233, 2013.

[42] Ahmed Eldawy and Mohamed F. Mokbel. Pigeon: A Spatial MapReduce Language. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1242–1245, Chicago, IL, March 2014.

[43] Ahmed Eldawy and Mohamed F. Mokbel. The Ecosystem of SpatialHadoop. *ACM SIGSPATIAL Special*, 6(3):3–10, 2014.

[44] Ahmed Eldawy and Mohamed F. Mokbel. SpatialHadoop: A MapReduce Framework for Spatial Data. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1352–1363, Seoul, South Korea, April 2015.

[45] Ahmed Eldawy, Mohamed F. Mokbel, Saif Alharthi, Abdulhadi Alzaidy, Kareem Tarek, and Sohaib Ghani. SHAHED: A MapReduce-based System for Querying and Visualizing Spatio-temporal Satellite Data. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1585–1596, Seoul, Korea, April 2015.

[46] Ahmed Eldawy, Mohamed F. Mokbel, and Christopher Jonathan. A Demonstration of HadoopViz: An Extensible MapReduce System for Visualizing Big Spatial Data. *Proceedings of the VLDB Endowment, PVLDB*, 8(12):1896–1907, 2015.

[47] Ahmed Eldawy, Mohamed F. Mokbel, and Christopher Jonathan. HadoopViz: A MapReduce Framework for Extensible Visualization of Big Spatial Data. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 601–612, Helsinki, Finland, May 2015.

[48] ESRI Tools for Hadoop. Available at `http://esri.github.io/gis-tools-for-hadoop/`, 2015.

[49] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining, KDD*, pages 226–231, Portland, OR, 1996.

[50] James Faghmous and Vipin Kumar. *Spatio-Temporal Data Mining for Climate Data: Advances, Challenges, and Opportunities*. Advances in Data Mining, Springer, 2013.

[51] James H. Faghmous, Yashu Chamber, Shyam Boriah, Frode Vikebø, Stefan Liess, Michel dos Santos Mesquita, and Vipin Kumar. A Novel and Scalable Spatio-Temporal Technique for Ocean Eddy Monitoring. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, pages 281–287, Toronto, Ontario, Canada, July 2012.

[52] James H. Faghmous, Matthew Le, Muhammed Uluyol, Vipin Kumar, and Snigdhansu Chatterjee. A Parameter-Free Spatio-Temporal Pattern Mining Model to Catalog Global Ocean Dynamics. In *Proceedings of the IEEE International Conference on Data Mining, ICDM*, pages 151–160, Dallas, TX, December 2013.

[53] Raphael A. Finkel and Jon Louis Bentley. Quad Trees: A Data Structure for Retrieval on Composite Keys. *Acta Informatica*, 4(1):1–9, 1974.

[54] Anthony Fox, Chris Eichelberger, James Hughes, and Skylar Lyon. Spatio-temporal Indexing in Non-relational Distributed Databases. In *Proceedings of the IEEE International Conference on Big Data*, pages 291–299, Santa Clara, CA, October 2013.

[55] GeoMesa. Available at `http://www.geomesa.org/`, 2015.

[56] GeoTrellis. Available at `http://geotrellis.io/`, 2015.

[57] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. The Google File System. In *Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles, SOSP*, pages 29–43, Bolton Landing, NY, October 2003.

[58] GnipBlog. Available at `https://blog.gnip.com/tag/geotagged-tweets/`, 2014.

[59] Google Maps APIs. Available at `https://developers.google.com/maps/`, 2016.

[60] Himanshu Gupta, Bhupesh Chawda, Sumit Negi, Tanveer A. Faruquie, L. V. Subramaniam, and Mukesh Mohania. Processing Multi-way Spatial Joins on Map-reduce. In *Proceedings of the International Conference on Extending Database Technology, EDBT*, pages 113–124, Genoa, Italy, March 2013.

[61] Ralf Hartmut Güting and Jiamin Lu. Parallel SECONDO: Scalable Query Processing in the Cloud for Non-Standard Applications. *ACM SIGSPATIAL Special*, 6(2):3–10, 2014.

[62] Antonin Guttman. R-Trees: A Dynamic Index Structure for Spatial Searching. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 47–57, Boston, MA, June 1984.

[63] Apache Hadoop. Available at `http://hadoop.apache.org/`, 2015.

[64] Lilian Harada, Miyuki Nakano, Masaru Kitsuregawa, and Mikio Takagi. Query Processing for Multi-Attribute Clustered Records. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 59–70, Queensland, Australia, August 1990.

[65] HBase. Available at `http://hbase.apache.org/`, 2015.

[66] Yaobin He, Haoyu Tan, Wuman Luo, Shengzhong Feng, and Jianping Fan. MR-DBSCAN: A Scalable MapReduce-based DBSCAN Algorithm for Heavily Skewed Data. *Frontiers of Computer Science*, 8(1):83–99, 2014.

[67] Telescope Hubbel Site: Hubble Essentials: Quick Facts. Available at `http://hubblesite.org/the_telescope/hubble_essentials/quick_facts.php`, 2015.

[68] Michael Isard, Mihai Budiu, Yuan Yu, Andrew Birrell, and Dennis Fetterly. Dryad: Distributed Data-Parallel Programs from Sequential Building Blocks. In *Proceedings of the ACM SIGOPS/EuroSys European Conference on Computer Systems*, pages 59–72, Lisbon, Portugal, March 2007.

[69] Ameet Kini and Rob Emanuele. Geotrellis: Adding Geospatial Capabilities to Spark. Available at `http://spark-summit.org/2014/talk/geotrellis-adding-geospatial-capabilities-to-spark`, 2014.

[70] Marcel Kornacker, Alexander Behm, Victor Bittorf, Taras Bobrovytsky, Casey Ching, Alan Choi, Justin Erickson, Martin Grund, Daniel Hecht, Matthew Jacobs, Ishaan Joshi, Lenni Kuff, Dileep Kumar, Alex Leblang, Nong Li, Ippokratis Pandis, Henry Robinson, David Rorke, Silvius Rus, John Russell, Dimitris Tsirogiannis, Skye Wanderman-Milne, and Michael Yoder. Impala: A Modern, Open-Source SQL Engine for Hadoop. In *Proceedings of the International Conference on Innovative Data Systems Research, CIDR*, Asilomar, CA, January 2015.

[71] Ravi Kothuri and Siva Ravada. Oracle spatial, geometries. In *Encyclopedia of GIS.*, pages 821–826. Springer, 2008.

[72] Scott T. Leutenegger, Mario A. Lopez, and Jeffrey M. Edgington. STR: A Simple and Efficient Algorithm for R-Tree Packing. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 497–506, Birmingham U.K., April 1997.

[73] Haojun Liao, Jizhong Han, and Jinyun Fang. Multi-dimensional Index on Hadoop Distributed File System. In *International Conference on Networking, Architecture, and Storage, ICNAS*, pages 240–249, Macau, China, July 2010.

[74] Ming-Ling Lo and Chinya V. Ravishankar. Spatial hash-joins. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 247–258, Montreal, Canada, June 1996.

[75] MODIS Land Products Quality Assurance Tutorial: Part:1. Available at `https://lpdaac.usgs.gov/sites/default/files/public/modis/docs/MODIS_LP_QA_Tutorial-1.pdf`, 2012.

[76] Jiamin Lu and Ralf Hartmut Güting. Parallel Secondo: Boosting Database Engines with Hadoop. In *International Conference on Parallel and Distributed Systems, ICPADS*, pages 738 –743, Singapore, December 2012.

[77] Jiamin Lu and Ralf Hartmut Güting. Parallel SECONDO: Practical and Efficient Mobility Data Processing in the Cloud. In *Proceedings of the IEEE International Conference on Big Data*, pages 17–25, Santa Clara, CA, October 2013.

[78] Jiamin Lu and Ralf Hartmut Güting. Parallel SECONDO: A Practical System for Large-Scale Processing of Moving Objects. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1190–1193, Chicago, IL, April 2014.

[79] Peng Lu, Gang Chen, Beng Chin Ooi, Hoang Tam Vo, and Sai Wu. ScalaGiST: Scalable Generalized Search Trees for MapReduce Systems. *Proceedings of the VLDB Endowment, PVLDB*, 7(14):1797–1808, 2014.

[80] Wei Lu, Yanyan Shen, Su Chen, and Beng Chin Ooi. Efficient Processing of k Nearest Neighbor Joins using MapReduce. *Proceedings of the VLDB Endowment, PVLDB*, 5(10):1016–1027, 2012.

[81] Qiang Ma, Bin Yang, Weining Qian, and Aoying Zhou. Query Processing of Massive Trajectory Data Based on MapReduce. In *International Workshop on Cloud Data Management, CloudDB*, pages 9–16, HongKong, China, October 2009.

[82] James MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, Berkeley, CA, June 1967.

[83] Amr Magdy, Louai Alarabi, Saif Al-Harthi, Mashaal Musleh, Thanaa Ghanem, Sohaib Ghani, and Mohamed F. Mokbel. Taghreed: A System for Querying, Analyzing, and Visualizing Geotagged Microblogs. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 163–172, Dallas, TX, November 2014.

[84] Amr Magdy, Louai Alarabi, Saif Al-Harthi, Mashaal Musleh, Thanaa M. Ghanem, Sohaib Ghani, Saleh Basalamah, and Mohamed F. Mokbel. Demonstration of Taghreed: A System for Querying, Analyzing, and Visualizing Geotagged Microblogs. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1416–1419, Seoul, South Korea, April 2015.

[85] James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers. Big data: The Next Frontier for Innovation, Competition, and Productivity. Technical report, McKinsey Global Institute, June 2011.

[86] Henry Markram. The Blue Brain Project. *Nature Reviews Neuroscience*, 7(2):153–160, 2006.

[87] Sergey Melnik, Andrey Gubarev, Jing Jing Long, Geoffrey Romer, Shiva Shivakumar, Matt Tolton, and Theo Vassilakis. Dremel: Interactive Analysis of Web-Scale Datasets. *Proceedings of the VLDB Endowment, PVLDB*, 3(1):330–339, 2010.

[88] Sergey Melnik, Andrey Gubarev, Jing Jing Long, Geoffrey Romer, Shiva Shivakumar, Matt Tolton, and Theo Vassilakis. Dremel: Interactive Analysis of Web-scale Datasets. *Communications of the ACM*, 54(6):114–123, 2011.

[89] Todd Mostak. An Overview of MapD (Massively Parallel Database). Technical report, Harvard University, Cambridge, MA, 2014.

[90] Jan Kristof Nidzwetzki and Ralf Hartmut Güting. Distributed SECONDO: A Highly Available and Scalable System for Spatial Data Processing. In *Proceedings of the International Symposium on Advances in Spatial and Temporal Databases, SSTD*, pages 491–496, Kong Kong, China, August 2015.

[91] Shoji Nishimura, Sudipto Das, Divyakant Agrawal, and Amr El Abbadi. $\mathcal{MD}$-HBase: A Scalable Multi-dimensional Data Infrastructure for Location Aware Services. In *Proceedings of the International Conference on Mobile Data Management, MDM*, pages 7–16, Luleå, Sweden, June 2011.

[92] Shoji Nishimura, Sudipto Das, Divyakant Agrawal, and Amr El Abbadi. $\mathcal{MD}$-HBase: Design and Implementation of an Elastic Data Infrastructure for Cloud-scale Location Services. *Distributed and Parallel Databases, DAPD*, 31(2):289–319, 2013.

[93] Open Geospatial Consortium. Available at `http://www.opengeospatial.org/`, 2015.

[94] Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. Pig Latin: A Not-so-foreign Language for Data Processing. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 1099–1110, Vancouver, BC, June 2008.

[95] Jignesh Patel and David DeWitt. Partition Based Spatial-Merge Join. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 259–270, June 1996.

[96] Jignesh M. Patel, Jie-Bing Yu, Navin Kabra, Kristin Tufte, Biswadeep Nag, Josef Burger, Nancy E. Hall, Karthikeyan Ramasamy, Roger Lueder, Curt J. Ellmann, Jim Kupsch, Shelly Guo, David J. DeWitt, and Jeffrey F. Naughton. Building a Scaleable Geo-Spatial DBMS: Technology, Implementation, and Evaluation. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 336–347, Tucson, AZ, May 1997.

[97] Linda Williams Pickle, Martha Szczur, Denise Lewis, , and David G. Stinchcomb. The Crossroads of GIS and Health Information: A Workshop on Developing a Research Agenda to Improve Cancer Control. *International Journal of Health Geographics*, 5(1):51, 2006.

[98] Gary Planthaber, Michael Stonebraker, and James Frew. EarthDB: Scalable Analysis of MODIS Data using SciDB. In *Proceedings of the ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data, BigSpatial*, pages 11–19, Redondo Beach, CA, November 2012.

[99] PostGIS. Available at `http://postgis.net/`, 2015.

[100] Jagan Sankaranarayanan, Hanan Samet, Benjamin E. Teitler, and Michael D. Lieand Jon Sperling. TwitterStand: News in Tweets. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 42–51, Seattle, WA, November 2009.

[101] Mohamed Sarwat. Interactive and Scalable Exploration of Big Spatial Data - A Data Management Perspective. In *Proceedings of the International Conference on Mobile Data Management, MDM*, pages 263–270, Pittsburgh, PA, June 2015.

[102] Sentinel-2. Available at `https://sentinel.esa.int/web/sentinel/missions/sentinel-2`, 2016.

[103] Shahed Web Interface. Available at `http://shahed.cs.umn.edu/`, 2015.

[104] Juwei Shi, Yunjie Qiu, Umar Farooq Minhas, Limei Jiao, Chen Wang, Berthold Reinwald, and Fatma Özcan. Clash of the Titans: MapReduce vs. Spark for Large Scale Data Analytics. *Proceedings of the VLDB Endowment, PVLDB*, 8(13):2110–2121, 2015.

[105] Konstantin Shvachko, Hairong Kuang, Sanjay Radia, and Robert Chansler. The Hadoop Distributed File System. In *IEEE Symposium on Mass Storage Systems and Technologies, MSST*, pages 1–10, Incline Villiage, NV, May 2010.

[106] Simple Feature Access standard by OGC. Available at `http://www.opengeospatial.org/standards/sfa`, 2015.

[107] James W. Stamos and Honesty C. Young. A symmetric fragment and replicate algorithm for distributed joins. *IEEE Transactions on Parallel and Distributed Systems, TPDS*, 4(12):1345–1354, 1993.

[108] Michael Stonebraker, Daniel J. Abadi, Adam Batkin, Xuedong Chen, Mitch Cherniack, Miguel Ferreira, Edmond Lau, Amerson Lin, Samuel Madden, Elizabeth J. O'Neil, Patrick E. O'Neil, Alex Rasin, Nga Tran, and Stanley B. Zdonik. C-Store: A Column-oriented DBMS. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, pages 553–564, Trondheim, Norway, August 2005.

[109] Michael Stonebraker, Paul Brown, Alex Poliakov, and Suchi Raman. The Architecture of SciDB. In *Proceedings of the International Conference on Scientific and Statistical Database Management, SSDBM*, pages 1–16, Portland, OR, June 2011.

[110] Michael Stonebraker, Paul Brown, Donghui Zhang, and Jacek Becla. SciDB: A Database Management System for Applications with Complex Analytics. *Computing in Science and Engineering*, 15(3):54–62, 2013.

[111] Farhan Tauheed, Laurynas Biveinis, Thomas Heinis, Felix Schürmann, Henry Markram, and Anastasia Ailamaki. Accelerating range queries for brain simulations. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 941–952, Washington D.C., April 2012.

[112] Yonette F. Thomas, Douglas Richardson, and Ivan Cheung. *Geography and Drug Addiction*. Springer Verlag, 2009.

[113] Ashish Thusoo, Joydeep Sarma Sen, Namit Jain, Zheng Shao, Prasad Chakka, Suresh Anthony, Hao Liu, Pete Wyckoff, and Raghotham Murthy. Hive: A Warehousing Solution over a Map-Reduce Framework. *Proceedings of the VLDB Endowment, PVLDB*, pages 1626–1629, 2009.

[114] Twitter. The About webpage. Available at `https://about.twitter.com/company`, 2015.

[115] Jacob VanderPlas, Emad Soroush, K. Simon Krughoff, and Magdalena Balazinska. Squeezing a Big Orange into Little Boxes: The AscotDB System for Parallel Processing of Data on a Sphere. *IEEE Data Engineering Bulletin*, 36(4):11–20, 2013.

[116] Hoang Vo, Ablimit Aji, and Fusheng Wang. SATO: A Spatial Data Partitioning Framework for Scalable Query Processing. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 545–548, Dallas, TX, November 2014.

[117] Huy T. Vo, Jonathan Bronson, Brian Summa, João Luiz Dihl Comba, Juliana Freire, Bill Howe, Valerio Pascucci, and Cláudio T. Silva. Parallel Visualization on Large Clusters using MapReduce. In *IEEE Symposium on Large Data Analysis and Visualization, LDAV*, pages 81–88, Providence, Rhode Island, October 2011.

[118] Skye Wanderman-Milne and Nong Li. Runtime Code Generation in Cloudera Impala. *IEEE Data Engineering Bulletin*, 37(1):31–37, 2014.

[119] Fusheng Wang, Ablimit Aji, and Hoang Vo. High Performance Spatial Queries for Spatial Big Data: From Medical Imaging to GIS. *ACM SIGSPATIAL Special*, 6(3):11–18, 2014.

[120] Guozhang Wang, Marcos Antonio Vaz Salles, Benjamin Sowell, Xun Wang, Tuan Cao, Alan J. Demers, Johannes Gehrke, and Walker M. White. Behavioral Simulations in MapReduce. *Proceedings of the VLDB Endowment, PVLDB*, 3(1):952–963, 2010.

[121] Kai Wang, Jizhong Han, Bibo Tu, Jiao Dai amd Wei Zhou, and Xuan Song. Accelerating Spatial Data Processing with MapReduce. In *International Conference on Parallel and Distributed Systems, ICPADS*, pages 229–236, Shanghai, China, December 2010.

[122] Randall T. Whitman, Michael B. Park, Sarah A. Ambrose, and Erik G. Hoel. Spatial Indexing and Analytics on Hadoop. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 73–82, Dallas, TX, November 2014.

[123] European XFEL: The Data Challenge. Available at `http://www.eiroforum.org/activities/scientific_highlights/201209_XFEL/index.html`, September 2012.

[124] Dong Xie, Feifei Li, Bin Yao, Gefei Li, Liang Zhou, and Minyi Guo. Simba: Efficient In-Memory Spatial Analytics. In *Proceedings of the ACM International Conference on Management of Data, SIGMOD*, pages 1071–1085, June 2016.

[125] Simin You, Jianting Zhang, and Le Gruenwald. Large-scale spatial join query processing in Cloud. In *International Workshop on Cloud Data Management, CloudDM, in Conjunction with the IEEE International Conference on Data Engineering, ICDE*, pages 34–41, Seoul, South Korea, April 2015.

[126] Jia Yu, Mohamed Sarwat, and Jinxuan Wu. GeoSpark: A Cluster Computing Framework for Processing Large-Scale Spatial Data. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems, ACM SIGSPATIAL*, pages 70:1–70:4, Seattle, WA, November 2015.

[127] Jie-Bing Yu and David J. DeWitt. Query Pre-Execution and Batching in Paradise: A Two-Pronged Approach to the Efficient Processing of Queries on Tape-Resident Raster Images. In *Proceedings of the International Conference on Scientific and Statistical Database Management, SSDBM*, pages 64–78, August 1997.

[128] Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauly, Michael J. Franklin, Scott Shenker, and Ion Stoica. Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing. In *Proceedings of the USENIX Conference on Networked Systems Design and Implementation, NSDI*, pages 15–28, San Jose, CA, April 2012.

[129] Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, and Ion Stoica. Spark: Cluster Computing with Working Sets. In *Proceedings of the USENIX Conference on Hot Topics in Cloud Computing, HotCloud*, volume 10, pages 10–16, Boston, MA, June 2010.

[130] Chi Zhang, Feifi Li, and Jeffrey Jestes. Efficient Parallel kNN Joins for Large Data in MapReduce. In *Proceedings of the International Conference on Extending Database Technology, EDBT*, pages 38–49, Berlin, Germany, March 2012.

[131] Shubin Zhang, Jizhong Han, Zhiyong Liu, Kai Wang, and Shengzhong Feng. Spatial Queries Evaluation with MapReduce. In *Proceedings of the International Conference on Grid and Cooperative Computing, GCC*, pages 287–292, Washington, DC, August 2009.

[132] Shubin Zhang, Jizhong Han, Zhiyong Liu, Kai Wang, and Zhiyong Xu. SJMR: Parallelizing spatial join with MapReduce on clusters. In *Proceedings of the IEEE International Conference on Cluster Computing and Workshops, CLUSTER*, pages 1–8, New Orleans, LA, August 2009.

[133] Xiangyu Zhang, Jing Ai, Zhongyuan Wang, Jiaheng Lu, and Xiaofeng Meng. An Efficient Multi-Dimensional Index for Cloud Data Management. In *International Workshop on Cloud Data Management, CloudDB*, pages 17–24, Hong Kong, China, 2009.

[134] Weizhong Zhao, Huifang Ma, and Qing He. Parallel K-Means Clustering Based on MapReduce. In *International Conference on Cloud Computing Technology and Science, CloudCom*, pages 674–679, Beijing, China, December 2009.

[135] Jingren Zhou, Nicolas Bruno, Ming-Chuan Wu, Per-Åke Larson, Ronnie Chaiken, and Darren Shakib. SCOPE: Parallel Databases Meet MapReduce. *The VLDB Journal*, 21(5):611–636, 2012.

[136] Jingren Zhou, Per-Åke Larson, and Ronnie Chaiken. Incorporating Partitioning and Parallel Plans into the SCOPE Optimizer. In *Proceedings of the International Conference on Data Engineering, ICDE*, pages 1060–1071, Long Beach, CA, March 2010.