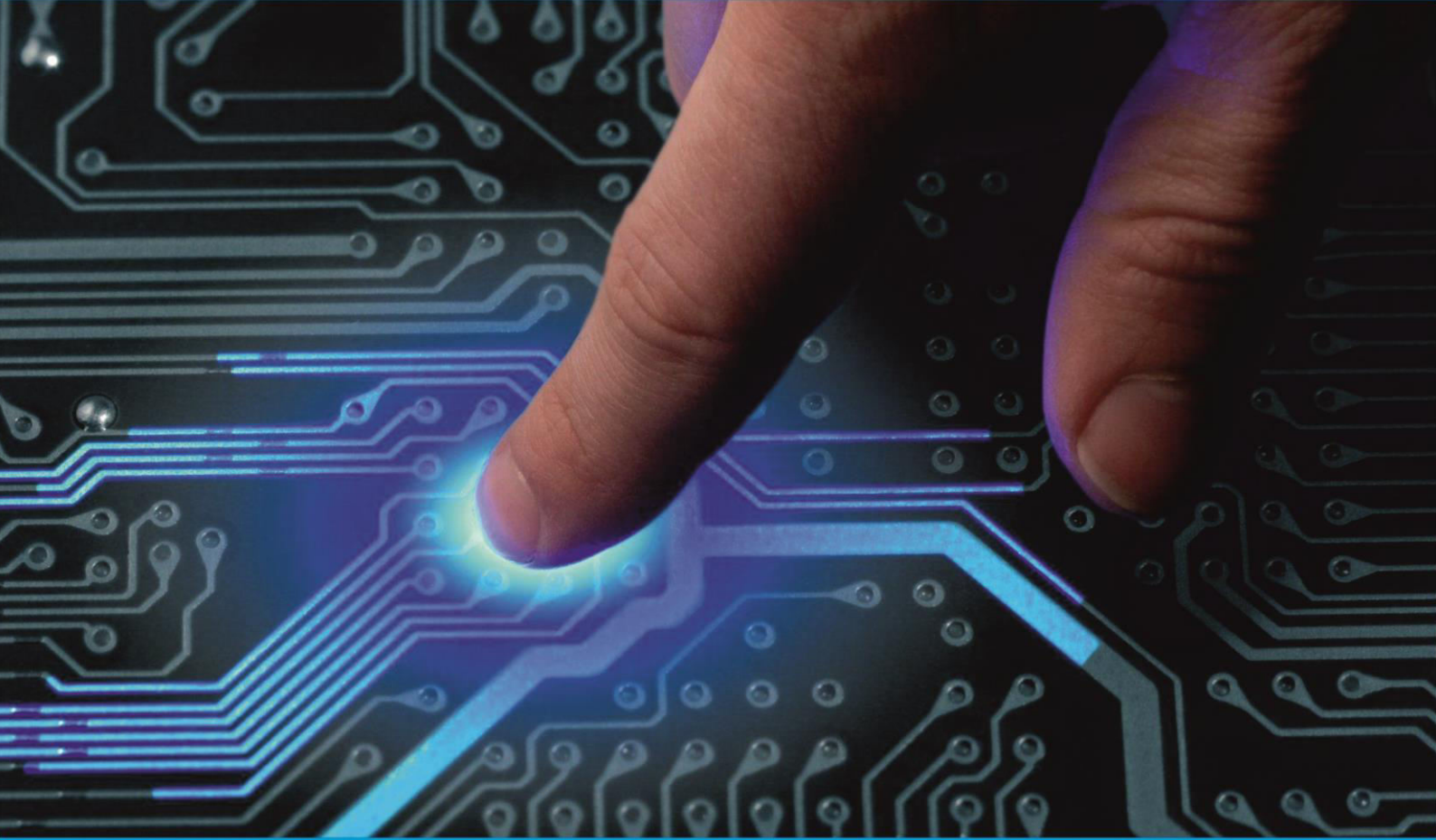




**IJIRCCCE**

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 3, March 2021

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

**Impact Factor: 7.488**

 9940 572 462

 6381 907 438

 [ijircce@gmail.com](mailto:ijircce@gmail.com)

 [www.ijircce.com](http://www.ijircce.com)

# Study on Conversion of Relational Databases to Big Data (VoltDB)

Ramu Vookanti<sup>1</sup>, E. Aravind<sup>2</sup>, M.Varaprasad<sup>3</sup>

Assistant Professor, Department of CSE, Chaitanya (Deemed to be University), Hanamkonda, Warangal Urban, Telangana, India<sup>1,2&3</sup>

**ABSTRACT:** We live in a data age, and there is a considerable measure of data streaming between systems, the internet, telephones, and other media. The data is being gathered and put away at remarkable rates.- The measure of accessible data has expanded by a gigantic sum in the previous few years in light of new friendly conduct and immense spread of social framework. Enormous data have assumed a vital part of development and growth. Relational database management systems (RDBMS), its utilization has been fundamentally reached out with new kinds of database systems. In particular, SQL has been discovered to be an incredible inquiry language in profoundly conveyed and versatile systems that cycle Big Data, i.e., datasets with high volume, speed and assortment. While conventional relational databases address just a tiny portion of the database systems scene, most database courses that cover SQL think about just the utilization of SQL regarding conventional relational systems.

**KEYWORDS:** Big data, SQL, Apache Hadoop

## I. INTRODUCTION

The measure of data that is being produced and put away is expanding dramatically step by step. It very well might be multiplying like clockwork, as per the gauge by IDC 2011. New progressed examination methods, and innovations permit clients to interface and question datasets. Significant data diagram advances and inventive procedures to disseminate, store, catch, oversee and analyze largesized datasets with high-speed and various constructions that conventional data management strategies are unequipped for handling. Increasingly huge quantities of inserted sensors, PDAs, PCs, and tablet PCs associated with the network produce tremendous data measures [1]. This data sets out new open doors to "separate more worth" for the zones that it is required. We have entered the period of "Big Data." Just as this data is created by individuals continuously, it may be analyzed progressively by high execution processing organizations, accordingly making improved dynamic. The International Data Corporation (IDC) accepts associations that are best ready to settle on realtime business choices utilizing Big Data arrangements will flourish.

In contrast, those that cannot embrace and utilize this move will progressively wind up at a severe disservice on the lookout and face likely disappointment. Several obstructions should have been defeated with the measure of data extending, for example, innovation challenges, association restrictions, and security/trust concerns [2]. While the measure of massive datasets is enthusiastically expanding, it likewise achieves many testing issues requesting actuate arrangements.

## II. BIG DATA OPERATIONAL AND ANALYTICAL TECHNOLOGIES

Even though Big Data may sound modern, it needs specific extraordinary innovations to effectively handle immense volumes of data in a decent range of time. Here is a portion of the innovations that can be applied to the treatment of big data. **Apache Hadoop** is a critical innovation used to deal with big data, examining and stream figuring [3]. Apache Hadoop is an open-source programming project that empowers the circulated handling of enormous data sets across production workers groups. It can be scaled up from a solitary worker to many machines and a severe adaptation to non-critical failure.

Rather than depending on top of the line equipment, these bunches' versatility comes from the product's capacity to recognize and deal with disappointments at the application layer. **Distributed file system** permits customer nodes to get to files through a PC organization. This way, several clients chipping away at different machines will want to share files and capacity assets. The customer nodes will not be ready to get to the square stockpiling; however, they can connect through a network convention [4]. This empowers a confined admittance to the file system relying upon the entrance records or on both servers and customers, which is again reliant on the protocol. **Data-Intensive Computing** is a class of similar processing applications that utilizes similar data to measure big data. This works dependent on the guideline of collocation of data, what is more, projects or calculations used to perform the algorithm. The equal and

distributed system between associated independent PCs cooperating as a solitary coordinated processing asset is utilized to measure/analyze big data[5]. **No SQL databases** - A few methodologies are received by NoSQL (Not Only SQL) for putting away and overseeing unstructured data. NoSQL databases separate data management and data stockpiling, while relational databases consolidate the two of them. One of the vital ideas of the NoSQL is to have the database centre around the errand of high-performance adaptable data stockpiling and give low-level admittance to a data management layer in a manner that permits data management undertakings to be written in the application layer rather than having data management rationale spread across in SQL[6].

### III. SQL IN NEWSQL SYSTEMS

NewSQL is a class of database systems that not just offers the same degrees of adaptability and accessibility of NoSQL systems yet likewise safeguards the ACID ensures, relational data model, and SQL inquiry language of conventional relational databases. Most NewSQL databases can learn SQL thinking about the application situations usually used to show conventional RDBMS. Nonetheless, a superior approach is to consider application situations that require the abilities of NewSQL systems, e.g., the need to prepare enormous or quick data in the financial exchange, web-based media networks, and web-based games[7]. Also, teachers ought to think about applications that exploit the highlights of the NewSQL system utilized in class, e.g., an application that requires handling a high number of short exchanges each second would be a solid match for VoltDB.

#### 1. Learning SQL with VoltDB

VoltDB is an in-memory and ACID-consistent NewSQL database system that utilizes a standard nothing engineering just as dividing and replication components to accomplish high exchange throughputs. VoltDB can be introduced on Linux and Mac OS X PCs and gives customer libraries to Java, Python, PHP, C++, C#, and so forth VoltDB can be utilized to learn numerous parts of SQL utilizing involved class exercises [8].

```

C1: md \data\db
C2: C:\mongodb\bin\mongod.exe
C3: C:\mongodb\bin\mongo.exe
C4: show dbs
C5: C:\mongodb\bin\mongoimport.exe -h <host:port> -d <your
database name> -c <collection> -u <dbuser> -p <dbpassword> --
type csv --file <path to .csv file> -headerline
C6: mongod://<dbuser>:<dbpassword>@<host:port>/<dbname>
Q1: SELECT N1 FROM "/project/taxes" WHERE ZIPCODE=85304
Q2: SELECT N2, ZIPCODE FROM "/project/taxes" WHERE STATE='NY'
Q3: SELECT ZIPCODE, A00100 FROM "/project/taxes" WHERE
STATE='NY' AND ZIPCODE!=0 ORDER BY (A00100) DESC LIMIT 10
Q4: SELECT DISTINCT STATE, AVG(A00200) FROM "/project/taxes"
WHERE ZIPCODE!=0 AND ZIPCODE!=99999 GROUP BY STATE

```

Figure 1: SlamData Commands and SQL Queries

Utilizing a VM is additionally a viable method to empower involved cooperation with NewSQL systems like VoltDB. For this situation, educators can utilize the VM accessible at VoltDB's site. The VM incorporates the introduced VoltDB programming and a bunch of test applications and instruments for application advancement. VoltDB can be utilized to learn numerous parts of SQL utilizing involved class exercises [9]. For example, it tends to gain proficiency with the SQL data definition and data control dialects. Utilizing a VM is additionally a viable method to empower an involved communication with NewSQL systems like VoltDB. For this situation, teachers can utilize the VM accessible at VoltDB's site[10]. The VM incorporates the introduced VoltDB programming and a bunch of test applications and devices for application advancement.



### Creating the Database

In VoltDB you define your database schema using SQL data definition language (DDL) statements just like other SQL databases. So, if we want to create a database table for the places where we live, the DDL schema might look like the following:

```
CREATE TABLE towns (  
town VARCHAR(64),  
county VARCHAR(64),  
state VARCHAR(2)  
);
```

The preceding schema defines a single table with three columns: town, county, and state. We could also set options, such as default values and primary keys. But for now we will keep it as simple as possible.

### Starting the Database and Loading the Schema

Once you have the schema defined, you can initialize and start the database, then load your schema. There are several options available when initializing a VoltDB database, which we will discuss later. But for now, we can use the simplest **init** and **start** commands to initialize and start the database using the default options on the current machine:

```
$ voltdbinit  
$ voltdb start
```

The **voltdbinit** command initializes a root directory that VoltDB uses to store its configuration, logs, and other disk-based information. You only need to initialize the root directory once for a production database. When doing development, where you often want to start over with new settings or a completely different schema, you can reuse the same root by using the **voltdbinit --force** command between runs. We will use both methods — starting a fresh database and restarting an existing database — in this tutorial. To start with, we will re-initialize the database root each time.

The **voltdb start** command tells VoltDB to start the database process. Once startup completes, the server reports the following message:

```
Server completed initialization.
```

Now you are ready to load your schema. To do that you use the VoltDB interactive command line utility, **sqlcmd**. Create a new terminal window and issue the **sqlcmd** command from the shell prompt:

```
$ sqlcmd  
SQL Command :: localhost:21212
```

The VoltDB interactive SQL command line first reports what database it has connected to and then puts up a numbered prompt. At the prompt, you can enter DDL statements and SQL queries, execute stored procedures, or type "exit" to end the program and return to the shell prompt.

To load the schema, you can either type the DDL schema statements by hand or, if you have them in a file, you can use the **FILE** directive to process all of the DDL statements with a single command. Since we only have one table definition, we can type or cut & paste the DDL directly into **sqlcmd**:

```
1. CREATE TABLE towns (  
2. town VARCHAR(64),  
3. county VARCHAR(64),  
4. state VARCHAR(2)  
5. );
```

### Using SQL Queries

Congratulations! You have created your first VoltDB database. Of course, an empty database is not terribly useful. So the first thing you will want to do is create and retrieve a few records to prove to yourself that the database is running as you expect.

VoltDB supports all of the standard SQL query statements, such as INSERT, UPDATE, DELETE, and SELECT. You can invoke queries programmatically, through standard interfaces such as JDBC and JSON, or you can include them in stored procedures that are compiled and loaded into the database.

But for now, we'll just try some ad hoc queries using sqlcmd. Let's start by creating records using the INSERT statement. The following example creates three records, for the towns of Billerica, Buffalo, and Bay View. Be sure to include the semi-colon after each statement.

1. `insert into towns values ('WARANGAL','TELANAGANA','INDIA');`
2. `insert into towns values ('VIZAG','ANDRAPRADESH','INDIA');`
3. `insert into towns values ('ROBE','OROMIA','ETHIOPIA');`

We can also use ad hoc queries to verify that our inserts worked as expected. The following queries use the SELECT statement to retrieve information about the database records.

```
select count(*) as total from towns;
TOTAL
-----
3
(1 row(s) affected)
select town, state from towns ORDER BY town;
TOWN          STATE
-----
WARANGAL     TELANAGANA
VIZAG        ANDRAPRADESH
ROBE         OROMIA
(3 row(s) affected)
```

When you are done working with the database, you can type "exit" to end the sqlcmd session and return to the shell command prompt. Then switch back to the terminal session where you started the database and press CTRL-C to end the database process.

#### IV. CONCLUSION

data management standards like physical and intelligent autonomy, explanatory questioning and cost-based improvement have driven to a multi-billion dollar industry. All the more critically, these technological advances have empowered the first round of business knowledge applications and established the framework for overseeing and investigating Big Data today. Many application situations require preparing exceptionally enormous datasets in a profoundly versatile and distributed style, and various kinds of Big Data systems have been intended to address this challenge. Once the innovations to analyze big data arrive at their pinnacle, it will get simpler for organizations to analyze massive datasets, recognize examples, and deliberately plan their moves depending on customer prerequisites that distinguished through memorable data.

#### REFERENCES

1. AnkitNarendrakumarSoni (2018). Data Center Monitoring using an Improved Faster Regional Convolutional Neural Network. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 7(4), 1849-1853.
2. YeshwanthValaboju, "AN OVERVIEW ON SAP FIORI DESIGN PRINCIPLES AND FIORI ARCHITECTURE FOR ANALYTICAL APPLICATIONS", The International journal of analytical and experimental modal analysis, Volume X, Issue IX, September 2018
3. Vishal DineshkumarSoni. (2018). IOT BASED PARKING LOT. International Engineering Journal For Research & Development, 3(1), 9. <https://doi.org/10.17605/OSF.IO/9GSA R>

4. BhagyaRekhaKalukurthi, “ IMPLEMENTATION OF BIG DATA ANALYTICS AND BIG DATA GOVERNANCE”, The International journal of analytical and experimental modal analysis, Volume VII, Issue I, May 2015.
5. Balnesridevi (2019). Review on challenges in SAAS model in cloud computing. Journal for innovative development in pharmaceutical and technical science, Volume-2, Issue-3 (March-2019). Page 8-11.
6. RakeshRojanala, “AN OVERVIEW ON CLOUD COMPUTING MODELS AND CLOUD DELIVERY MODELS”, The International journal of analytical and experimental modal analysis, Volume IV, Issue I, JAN-JUNE 2012.
7. JubinDipakkumar Kothari (2018). A Case Study of Image Classification Based on Deep Learning Using Tensorflow International Journal of Innovative Research in Computer and Communication Engineering, Vol. 6, Issue 4, April 2018, 3888-3892.
8. BhagyaRekhaKalukurthi, “A Comprehensive Review on Machine Learning and Deep Learning”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 8, Issue 6, June 2019.
9. JubinDipakkumar Kothari (2018). Detecting Welding Defects in Steel Plates using Machine Learning and Computer Vision Algorithms, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 7, Issue 9, September 2018, 3682-3686.
10. AnithaEemani, “Achieving Network Security and Security Mechanisms at Networking Layers”, International Journal of Information Technology and Management”, Vol. 11, Issue No. 17, November-2016.
11. PothugantiKarunakar, JagadishMatta, R. P. Singh, O. Ravi Kumar, (2020), Analysis of Position Based Routing Vanet Protocols using Ns2 Simulator, International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-9 Issue-5, March 2020.
12. Soni, Vishal Dinesh kumar and Soni, AnkitNarendrakumar and pothuganti, karunakar, Student Body Temperature and Physical Distance Management Device in Classroom Using 3D Stereoscopic Distance Measurement (2020). International Journal of Innovative Research in Science Engineering and Technology 9(9):9294-9299 (2020)
13. KetulkumarGovindbhaiChaudhari. (2019). Review on Challenges and Advanced Research Areas in Internet of Things. International Journal of Innovative Research in Computer and Communication Engineering, 7(7), 3570-3574. DOI: 10.15680/IJIRCE.2019. 0707016.
14. Soni, Vishal Dineshkumar, Role of AI in Industry in Emergency Services (2018). International Engineering Journal For Research & Development, 3(2), 6. <https://doi.org/10.17605/OSF.IO/C67BM>
15. YeshwanthValaboju, “A LITERATURE REVIEW ON NEURAL NETWORK ARCHITECTURES”, Journal of Interdisciplinary Cycle Research, Volume VII, Issue II, December 2015
16. Soni, AnkitNarendrakumar, Diabetes Mellitus Prediction Using Ensemble Machine Learning Techniques (July 3, 2020). Available at SSRN: <https://ssrn.com/abstract=3642877> or <http://dx.doi.org/10.2139/ssrn.3642877>.
17. Balnesridevi (2015), Recovery of Data in Cluster Computing By Using Fault Tolerant Mechanisms, IOSR Journal of Computer Engineering (IOSR-JCE), Volume 17, Issue 1, Ver. II (Jan – Feb. 2015), PP 40-45.
18. AnithaEemani, “A Comprehensive Review on Network Security Tools”, Journal of Advances in Science and Technology, Vol. 11, Issue No. 22, May-2016
19. KetulkumarGovindbhaiChaudhari. (2019). Windmill Monitoring System Using Internet of Things with Raspberry Pi. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 8(2), 482-485. DOI:10.15662/IJAREEIE.2019.0802043.
20. AnkitNarendrakumarSoni (2018). Smart Devices Using Internet of Things for Health Monitoring. International Journal of Innovative Research in Science, Engineering and Technology, 7(5), 6355-6361. doi:10.15680/IJIRSET.2018.0705233
21. JubinDipakkumar Kothari (2018). Garbage Level Monitoring Device Using Internet of Things with ESP8266, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 7, Issue 6, June 2018, 2995-2998.
22. KetulkumarGovindbhaiChaudhari. (2019). Water Quality Monitoring System using Internet of Things and SWQM Framework. International Journal of Innovative Research in Computer and Communication Engineering, 7(9), 3898-3903. DOI: 10.15680/IJIRCE.2019. 0709008
23. RakeshRojanala, “Cloud Computing Characteristics and Deployment of Big Data Analytics in The Cloud”, International Journal of Scientific Research in Science and Technology, Volume VIII, Issue II, March-April 2014.
24. Vishal DineshkumarSoni. (2019). IOT connected with e-learning. International Journal on Integrated Education, 2(5), 273-277. <https://doi.org/10.31149/ijie.v2i5.496>



INNO  SPACE  
SJIF Scientific Journal Impact Factor

Impact Factor:  
7.488

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 9940 572 462  6381 907 438  [ijircce@gmail.com](mailto:ijircce@gmail.com)



[www.ijircce.com](http://www.ijircce.com)

Scan to save the contact details