

広域センサネットワークのための Geohash を用いた時空間分散データベース

末松 直樹† 馬場 大輔† 宮崎 敏明†

会津大学大学院コンピュータ理工学研究科†

1. 序論

センサネットワークは物理環境の観測に広く普及し、その観測データの蓄積にはリレーショナルデータベース (RDB) や時系列データベース等、様々なデータベースが用いられている。しかし、広域に展開されたセンサネットワークが生成するデータを管理するデータベース (DB) では、非効率なネットワークトラフィックの抑制や、ネットワーク分断に対する耐性等、留意すべき点がある。また、HBase[1] や Cassandra[2] 等の key-value 型 DB は一要素でしか範囲検索が出来ないという制約を持つため、地理空間の範囲検索と時系列の範囲検索を同時に行うことは難しい。本稿では、多数の独立したセンサネットワークを統合したシステムにおいて、Geohash[3] を用いることで地理空間と時系列の同時範囲検索を可能とする分散 DB システムを提案する。また実際に計 100 万件/秒のデータ蓄積が可能であることを示す。

2. 広域センサネットワークと DB に対する要求

ここでは、複数の独立したローカルセンサネットワークをインターネット等の広域ネットワークを介して接続し、シームレスに各地のセンサデータを取得可能なシステムを想定する。本システムで用いる DB において留意すべき点を列挙する。

- 地理的範囲の拡張性: システム稼働後でも新たなローカルセンサネットワークをシステムへ追加出来ること
- 保存可能容量の拡張性: システム稼働中でも、システムを止めることなくストレージ容量を増やせること
- ネットワーク分断耐性: 想定するシステムは地理的に広域なシステムなためネットワーク分断が懸念されるが、その場合でもデータを保存し続けられること
- 十分な書き込みパフォーマンス: システム全体の単位時間当たりのセンシングデータ生成件数はシステム規模に比して大きくになるため、そのデータを保存出来るだけの十分な書き込み速度が確保できること

これらの留意事項を念頭に、我々は次に示す DB システムを提案する。

3. 提案 DB システム

想定する広域センサネットワークで用いるための DB の構築方針としては、大まかに以下の 2 つが考えられる。

- 集中保存型: 1 箇所又は数カ所の大規模なデータセンタで全ローカルセンサネットワークのデータを管轄する。(図 1)

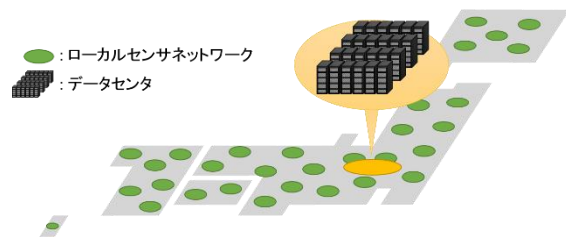


図 1. 集中保存型

- 分散ローカル保存型: 地区毎に DB を置き、その地区のデータはその地区の DB が管轄する。(図 2)

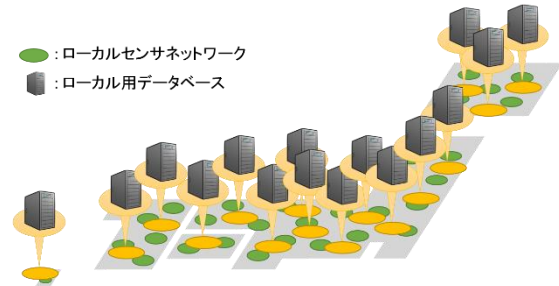


図 2. 分散ローカル保存型

集中保存型はマシン間のネットワーク遅延が小さいこと等から、Hadoop[4]などのマスタ・スレーブ型の分散 DB の活用が考えられる。Hadoop やその上で動作する HBase を用いた時空間 DB の先行研究としては、MD-HBase[5]や Spatial Hadoop[6]等がある。しかし集中保存型は、ネットワーク分断があった際データを保存し続けられない問題がある。

分散ローカル保存型で最も直感的な実現方法としては、MySQL や PostgreSQL 等の RDB を各地区に配置し、それらを統合するメカニズムを用意する方法が考えられる。しかし、拡張性をどのように確保するか等、様々な問題を解決しつつ、その統合メカニズムを実現することは容易ではない。そこで我々は、P2P ネットワーク技術に基づく単一障害点を持たない key-value 型分散 DB である Cassandra を用いた、分散ローカル保存型 DB システムの実現方法を提案する。提案方法の要点を以下に列挙する:

Space-Time Distributed Database Using Geohash for Wide-area Sensor Networks

†Naoki Suematsu, †Daisuke Baba, †Toshiaki Miyazaki

†Graduate School of Computer Science and Engineering, The University of Aizu

●非分散保存: Cassandra において、データの保存先マシンはカスタマイズ可能な関数によって決まる。その関数に適切なものを用いることで、文字列データである key が前方一致するデータを1つのマシンにまとめて保存することが出来る。

●Geohash を用いた地区ごとの保存: Geohash は緯度経度の範囲(グリッド)情報を1つの文字列で表すコーディング方法の一つで、その文字列が短いほど広範囲を表し、長いほど狭い範囲を表す。前述の非分散保存と Geohash を合わせて用いることで、Geohash の示すグリッド内のデータを一つのマシンにまとめて保存することが可能となる。よって、ネットワーク分断があった際にも近傍の Cassandra を起動したマシン(以下ピア)に保存し続けることが可能である。また Cassandra 等の key-value 型 DB は一要素でしか範囲検索が出来ないという制約を持つため、時間範囲指定が必須な時系列データの検索に地理的範囲の検索を同時に行うことは難しいが、書き込む際に1件のデータに対して Geohash の長さを変えた複数パターンのデータを書き込むことでグリッドに基づく地理的な範囲検索が可能となる。

●時系列データのシーケンシャルアクセス: Cassandra では、データ内のある要素でソートしてディスクへ保存することが出来る。よって、時間軸での範囲検索はシーケンシャルなディスクアクセスとすることが出来、高速化が見込まれる。図3にCQL(Cassandra Query Language)形式でテーブル構造を示す。

```
CREATE TABLE dasn.sensorlog (
  geohash_sensortype_date text,
  timestamp_tags text,
  sensor_value double,
  PRIMARY KEY (geohash_sensortype_date,
               timestamp_tags)
) WITH CLUSTERING ORDER BY (timestamp_tags DESC)
```

図3. テーブル構造

4. エミュレーションネットワーク上での拡張性検証

前項までに示したDBシステムを構築し、ピア数とネットワーク遅延の影響を検証した。検証では、同性能のマシンを150台用意し、ピア数を1, 5, 10, 15, ..., 150台と変えCassandraクラスタを構築した。全てのピアはL2ネットワークで繋げた。またピア全てに書き込みクライアントを起動し、ローカルのピアに毎秒25,200件のデータを書き込み所要時間を計測し、全書き込みクライアントの全試行の平均値を得た。25,200件は、4種のセンサを搭載した900台のセンサノードが1秒間隔でセンシングする状態で、Geohashの長さパターンを6~12の7パターンとしたデータ件数である。また、COREネットワークエミュレータ[7]で構築した広域エミュレーションネットワーク上でも同様の検証を行った。

COREは物理ネットワークデバイスを仮想エミュレーションネットワークに接続可能な、リアルタイムネットワークエミュレータである。エミュレーションネットワークのトポロジはピア間の最大ルータ間リンク数が6のツリー状とし、全てのルータ間に与える遅延を5msと50msの2パターンとした。結果を図4に示す。

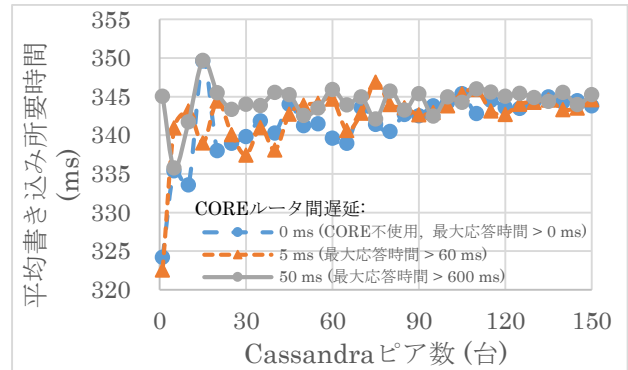


図4. 検証結果

検証結果は、ピア間のネットワーク遅延は書き込み速度に影響しないこと、及びCassandraクラスタ内ピア数は書き込み速度に大きな影響を与えないことを示した。また、実際に計300万件/秒以上のデータをストア可能なことを確認した。

5. 結論

本稿では、複数のローカルセンサネットワークを統合して一つのシステムとする広域センサネットワークに最適なデータベースシステムを提案し、ネットワークエミュレータと150台のマシンを用いその拡張性を確認した。

謝辞 本研究の一部は、総務省戦略的情報通信研究開発推進制度(SCOPE No. 121802001)の支援を受けて実施したものである。また、実験に際しては、JGN-X/StarBED³を用いた。

参考文献

[1] George Lars, "HBase: The Definitive Guide," 1st edition,
 [2] Prashant Malik, Avinash Lakshman. "Cassandra - a decentralized structured storage system" The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware (LADIS 09), October 2009.
 [3] "Geohash", <http://www.geohash.org>
 [4] T. White, "Hadoop: The Definitive Guide," O'Reilly Media, Inc., 2009.
 [5] S. Nishimura, S. Das, D. Agrawal and A. Ei Abbadi "MD-HBase: design and implementation of an elastic data infrastructure for cloud-scale location services", Distributed and Parallel Databases, vol. 31, no. 2, pp.289-319 2013
 [6] Ahmed Eldawy and Mohamed F. Mokbel, "SpatialHadoop: A MapReduce Framework for Spatial Data", In Proceedings of the IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April, 2015
 [7] J. Ahrenholz, C. WA Danilov, T. R. Henderson, J. H. Kim, "CORE: A real-time network emulator," IEEE Military Communications Conference (MILCOM 2008), pp. 1-7, 2008.