A complex technical background graphic featuring a central circular structure resembling a server or a data center, overlaid with various digital elements such as binary code (0s and 1s), circuit traces, and a waveform. The overall aesthetic is futuristic and data-oriented.

ibaHD-Server Benchmark

Vergleich zwischen ibaHD-Server und
zeitbasierten Datenbanksystemen

White Paper
Ausgabe 1.1

Messsysteme für
Industrie und Energie

Herausgeber

iba AG
Königswarterstr. 44
90762 Fürth
Deutschland

Kontakte

Zentrale +49 911 97282-0
Telefax +49 911 97282-33
Support +49 911 97282-14
Technik +49 911 97282-13
E-Mail iba@iba-ag.com
Web www.iba-ag.com

© iba AG 2021, alle Rechte vorbehalten.

Autor

Fabian Mielke
Software Developer

Ausgabe	Datum	Autor	Änderungen
1.1	07-2021	FM	Performance-Vergleich

Inhalt

1	Einleitung	4
2	Vergleich ibaHD-Server mit InfluxDB und TimescaleDB	5
2.1	Technischer Hintergrund	5
2.2	Ausrichtung.....	6
2.3	Performance	7
3	Anwendungsszenarien	9
4	Zukünftige Entwicklungen	10
4.1	Skalierbarkeit	10
4.2	Verfügbarkeit	10
4.3	Performance-Optimierungen.....	10
5	Glossar	11
6	Referenzen	12
7	Support und Kontakt	14

1 Einleitung

Auf dem Markt existiert eine Vielzahl von Datenbanksystemen mit speziellen Architekturen für unterschiedliche Aufgaben. Einige Datenbanksysteme sind spezialisiert auf die Speicherung und Verwaltung von zeitbezogenen Daten. Sie zeichnen sich im Allgemeinen dadurch aus, dass Zeitstempel als primäre Schlüssel der Datentabellen verwendet werden.

iba hat im Rahmen eines internen Projekts einen Vergleich zwischen ibaHD-Server und den zwei zeitbasierten Datenbanksystemen InfluxDB und TimescaleDB durchgeführt. Die beiden Systeme sind bekannte Stellvertreter der Zeitreihendatenbanken [1]. Die Zielsetzung war ein transparenter Vergleich der vorhandenen Features und der Performance. Die Ergebnisse und insbesondere die wesentlichen Unterschiede zwischen den Systemen werden in diesem White Paper dargestellt.

Benchmarks unterliegen immer bestimmten Randbedingungen und sind deswegen schwer miteinander zu vergleichen. Vor allem, wenn sich die verglichenen Systeme in ihrem idealen Anwendungsszenario unterscheiden. Um einen objektiven Benchmark durchzuführen, müssen diese Szenarien mit in Betracht gezogen werden. Im Folgenden ist die Zusammenfassung der Ergebnisse des Benchmarks dargestellt.

2 Vergleich ibaHD-Server mit InfluxDB und TimescaleDB

Den drei Datenbanksystemen liegen verschiedene technische Ansätze zugrunde. InfluxDB ist eine schemafreie Zeitreihendatenbank, d. h. es muss im Vorfeld kein Tabellenschema angelegt werden. Stattdessen gruppiert InfluxDB die Datensätze anhand von *Tags* ein, die dem Datensatz hinzugefügt werden können. Mehrere Datensätze, die exakt die gleiche Kombination an Tags haben, gehören zu einer Zeitreihe. Dieser Ansatz wird auch als Tagset-Modell bezeichnet. Die Daten werden mit dem Line Protocol als Plain-Text-Format an InfluxDB gesendet. Im Gegensatz dazu verfolgt TimescaleDB ein Hybridkonzept aus relationalen und zeitbasierten Datenbanken [2, 3]. Es kann als Erweiterung zu PostgreSQL installiert werden. Die Zeitreihen sind in TimescaleDB äußerlich fast nicht von normalen relationalen Tabellen zu unterscheiden. Im Grunde handelt es sich dabei um abstrakte Hypertabellen, die Zeitreihen als sogenannte *Chunks* speichern. Mit TimescaleDB können relationale und Zeitreihendaten miteinander kombiniert werden.

Mit ibaHD-Server wird eine andere Strategie verfolgt, mit der Messdaten vergleichsweise deutlich schneller und effizienter gespeichert werden können. Die Effizienz wird durch eine schnelle und verlustfreie Encodierung erreicht. Insbesondere bei der hochfrequenten Datenerfassung hat die Speichereffizienz höchste Priorität. ibaHD-Server bietet eine nahtlose Integration von iba-Daten in das iba-System und ist nicht auf leistungsstarke Hardware angewiesen.

2.1 Technischer Hintergrund

ibaHD-Server im Vergleich mit InfluxDB und TimescaleDB.

Interfaces	ibaHD-Server	InfluxDB	TimescaleDB
Datenaufzeichnung	ibaPDA ibaDatCoordinator ibaHD-Server Import	HTTP API UDP (JSON) Client Bibliotheken	ADO.NET JDBC Native C Library ODBC Streaming API
Lesezugriff	ibaAnalyzer ibaDatCoordinator ibaDaVIS gRPC-API	HTTP API UDP (JSON) Client Bibliotheken (Wrapper)	ADO.NET JDBC Native C Library ODBC Streaming API
Query Language	-	InfluxQL FluxQL	SQL

2.2 Ausrichtung

Die Intention von TimescaleDB und InfluxDB unterscheiden sich von der des ibaHD-Servers. Ein wesentlicher Bestandteil der zwei Systeme ist die Retention Policy, die im weitesten Sinne eine Aufräumstrategie darstellt. Es kann beispielsweise festgelegt werden, dass veraltete Daten gelöscht werden oder ein *Downsampling* erfolgt. Ein Argument von TimescaleDB hierfür ist, dass die Relevanz von älteren Daten mit der Zeit abnimmt, weil diese nicht mehr aktualisiert werden [4]. Mit dem Downsampling geht allerdings auch der Informationsgehalt im Detail verloren.

ibaHD-Server verfolgt in dieser Hinsicht eine andere Strategie. Um Messdaten auch für Dokumentations- und Analysezwecke in originaler Auflösung zu halten, werden die Messdaten sofort und verlustfrei komprimiert. Eine Limitierung des verwendeten Speicherplatzes ist dennoch konfigurierbar. Die Aggregation erfolgt zusätzlich, sodass bei Bedarf auch Langzeitanalysen ausgeführt werden können, die nicht die Originalauflösung benötigen. Auf diese Weise kann schnell ein Überblick über die aufgezeichneten Daten gewonnen werden. Für die Detailanalyse können dann die Rohdaten herangezogen werden.

	ibaHD-Server	InfluxDB	TimescaleDB
Strategie	Hohe Schreibgeschwindigkeit Stabile Übertragung zum Server durch <i>ein Layered Storage</i> Konzept mit einem Zwischenspeicher	Vorhalten aktueller Datensätze im Arbeitsspeicher Abspeichern älterer Daten auf der Festplatte	Vorhalten aktueller Daten-Chunks im Arbeitsspeicher Abspeichern älterer Daten auf der Festplatte
Aggregation	Aggregation erfolgt zusätzlich zu den Originaldaten Dauerhafte Verfügbarkeit der Daten in ihrer originalen Auflösung für Dokumentationszwecke <i>Retention Policy</i> mit optionalem Speicherlimit	<i>Retention Policy</i> ermöglicht das automatische Löschen älterer Daten <i>Downsampling</i> [5]	<i>Retention Policy</i> ermöglicht das automatische Löschen älterer Daten Kontinuierliche Aggregation durch <i>Downsampling</i> [4]
Ressourcen	Automatisches Ressourcenmanagement Optionale Limitierung der zu verwaltenden Datenmenge	Konfiguration bzw. Limitierung der verwendeten Ressourcen möglich [6]	Konfiguration bzw. Limitierung der verwendeten Ressourcen möglich [7]
Zeitreihenbezogene Metadaten	Metadaten sind vollständig in den Messdatensegmenten enthalten	Kein relationales Konzept für Metadaten [8]	Relation zwischen Zeitreihen und Metadaten möglich [9]

	ibaHD-Server	InfluxDB	TimescaleDB
Nicht-äquidistante Daten	Nur äquidistante Zeitreihen möglich <i>Event Store</i> unterstützt nicht-äquidistante Datensätze	Nicht-äquidistante Zeitreihen werden nativ unterstützt	Nicht-äquidistante Zeitreihen werden nativ unterstützt
Skalierbarkeit	Limitiert	Skalierbar in einem Cluster mit <i>Influx Enterprise</i> <i>InfluxDB Cloud</i> als Service bei <i>Amazon Web Services</i> , der <i>Google Cloud Platform</i> und <i>Microsoft Azure</i> verfügbar [10]	Skalierbares Cluster bei <i>Amazon Web Services</i> , der <i>Google Cloud Platform</i> und bei <i>Microsoft Azure</i> verfügbar [11]
Zusätzliche Features	Verwaltung und Speicherung von Ereignissen in einem separaten <i>Event Store</i> möglich Demo-Clients verfügbar für die ibaHD-Server API über gRPC (Python, C++, C#, Java)	<i>Telegraph</i> Software für das Sammeln von Messdaten und für Berichte Client-Bibliotheken verfügbar	<i>PostgreSQL</i> -Funktionalitäten sind kompatibel mit TimescaleDB
Betriebssystem	Windows	Linux OS X Windows (kein offizieller Support)	Linux OS X Windows

2.3 Performance

Ein wichtiger Bestandteil des Benchmarks war neben den Features die Untersuchung der Schreib- und Leseperformance. Für diesen Zweck wurden auf einem dedizierten Computer Daten generiert, die hinsichtlich ihrer Struktur, ihrer Frequenz und ihres Mengengerüsts typischen messtechnischen Anwendungsszenarien entsprechen. Die Datenbanksysteme wurden mithilfe einer eigens entwickelten Software getestet. Die Testsoftware generiert Datensätze, die an einen Client übergeben werden. Dieser Client formatiert die Daten und bereitet sie für die Schnittstelle der Datenbank vor. Anschließend werden die Datensätze eingefügt. Auf diese Weise werden gleiche Ausgangsbedingungen geschaffen.

Im Vorfeld wurden alle drei Systeme optimal konfiguriert. Als Vergleichsmetrik für die Lese- und Schreibgeschwindigkeit wurde die Einheit *metrics per second* gewählt. Die Struktur einer solchen *Metrik* ist an typische Messdaten angelehnt, d. h. sie besteht aus einem Zeitstempel,

einem Signalnamen und einer Gleitkommazahl, die den Messwert symbolisiert. In der gemessenen Zeit sind bei allen drei Systemen ebenfalls die nötigen Konvertierungen für die Schnittstelle der jeweiligen Systeme enthalten. Bei ibaHD-Server ist dies beispielsweise die Enkodierung, bei TimescaleDB das Einfügen in SQL-Befehle und bei InfluxDB die Formatierung in das Line Protocol.

Als Testcomputer wurde ein Desktopcomputer mit einem Intel Xeon E-2146G 3.50GHz-Prozessor mit 6 Kernen und 16 GB Arbeitsspeicher verwendet. Die Versionen der Datenbanksysteme waren ibaHD-Server 2.5.0, InfluxDB 1.8.3 und TimescaleDB 2.0 mit PostgreSQL 12 64-bit. Alle Systeme wurden auf Windows 10 getestet.

Sowohl in der Schreib- als auch in der Lesegeschwindigkeit übertrifft ibaHD-Server die anderen beiden Zeitreihendatenbanken bei weitem. Während bei ibaHD-Server die Daten laufend kodiert werden können, wird bei InfluxDB und bei TimescaleDB die Performance durch die Schnittstelle und die internen Datenbankmechanismen reduziert. Die folgenden Ergebnisse sind Mittelwerte mehrerer Testreihen.

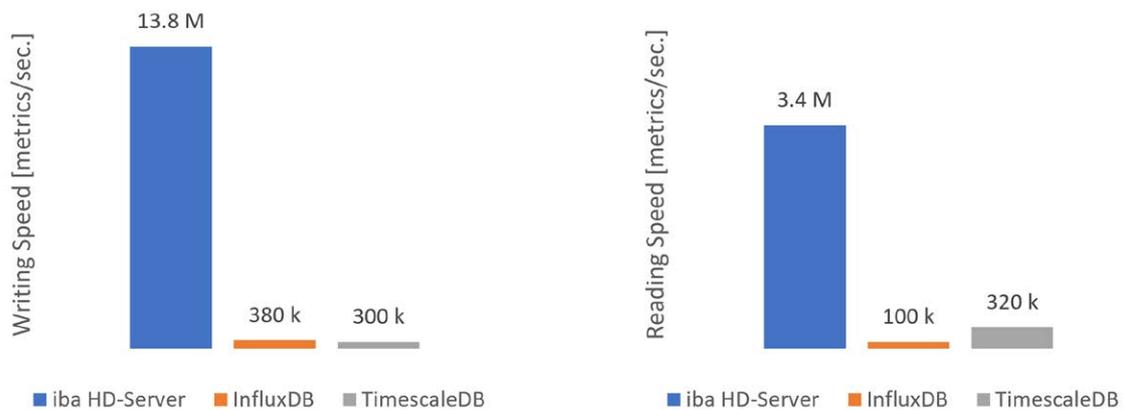


Abb. 1: Mittelwerte der Schreib- und Lesegeschwindigkeit

Die Durchführung von vergleichbaren Benchmarks ist ein aufwändiges Verfahren, das vielen Einflussparametern unterliegt. Um Fehler und Irrtümer auszuschließen, wurden die Ergebnisse mit anderen Benchmarks und Veröffentlichungen validiert [12, 13, 14, 15]. In diesen Vergleichsquellen wurden ähnliche Kennzahlen für InfluxDB und TimescaleDB ermittelt.

3 Anwendungsszenarien

Alle getesteten Datensysteme, einschließlich ibaHD-Server, eignen sich jeweils für bestimmte Anwendungen am besten. Die nachfolgende Tabelle fasst diese Szenarien zusammen.

ibaHD-Server	InfluxDB	TimescaleDB
Hochfrequentes Aufzeichnen von Messdaten Verlustfreie Datenaufzeichnung Langzeitverfügbarkeit von Prozess- und Produktionsdaten für Analysen und Produktdokumentation Integration in das iba-System <i>Echtzeit</i> -Visualisierung von Messdaten mit minimaler Verzögerung	Monitoring Anwendungen Industrial Internet of Things Anwendungen	Anwendungen, bei denen relationale Strukturen erforderlich sind Anwendungen mit einer großen Anzahl verschiedener Zeitserien [16]

4 Zukünftige Entwicklungen

4.1 Skalierbarkeit

ibaHD-Server ist geeignet für viele Daten der gleichen Struktur. In der Messtechnik ist es üblich, analoge und digitale Signale in einer Baumstruktur mit Modulen zu gliedern. Die Zuordnung zwischen den Signalen der Eingangsquellen muss einmalig vor dem laufenden Betrieb der Software durchgeführt werden und wird als Konfiguration gespeichert. Eine hohe Varianz der Datenstruktur, z. B. durch sich wechselnde Signalkonfigurationen, reduziert die Leistungsfähigkeit stark. Auch Lösungen, in der eine Vielzahl von Clients simultan Daten an ibaHD-Server zur Abspeicherung senden, stellt für die Infrastruktur eine große Herausforderung dar. In Zukunft wird die iba AG ebenfalls ein skalierbares Konzept, z. B. für IoT-Anwendungen, bieten.

4.2 Verfügbarkeit

Mit einem skalierbaren ibaHD-Server-System als cluster- oder cloudfähigem Dienst kann auch die Verfügbarkeit der Messdaten gesteigert werden. Fällt beispielsweise ein *Datenknoten* aus, können andere Instanzen die angefragten Daten zurückliefern. Diese Fähigkeit wird durch Replikation erreicht. Neben der Redundanz kann mit einem solchen System auch die Auslastung durch Lese- und Schreibzugriffe zwischen den Knoten verteilt werden.

4.3 Performance-Optimierungen

ibaHD-Server erreicht bereits hohe Datengeschwindigkeiten auf einfachen Desktopcomputern und arbeitet aufgrund der laufenden Encodierung sehr effizient. Dennoch verbessert iba die Systeme kontinuierlich.

5 Glossar

Aggregation	Verdichtung oder Komprimieren von Informationen, z. B. durch Verfahren zur Mittelwertbildung oder durch <i>Downsampling</i> .
Chunk	Eine definierte Menge an Datensätzen oder Informationen, die Teil der Gesamtdatenmenge sind.
Cluster	Vernetzung von Computern, um z. B. die Ausfallsicherheit eines Systems zu garantieren oder die Gesamtrechenleistung eines Systems zu erhöhen.
Datenknoten	Teil eines Verbunds aus mehreren Computern (<i>Cluster</i>), der einen Teil der gesamten Daten speichert.
Downsampling	Systematische Reduktion an Datensätzen, z. B. durch Löschen jedes zehnten Datensatzes.
Echtzeitfähigkeit	Fähigkeit eines Systems, garantiert innerhalb einer definierten Zeit zu reagieren. Die Echtzeitfähigkeit sagt nichts über die Leistung oder Geschwindigkeit eines Systems aus.
Metrik	Struktur und Informationen, die einen Datensatz definieren. Eine Metrik besteht z. B. aus einem Zeitstempel, einem <i>Tag</i> und einer Zahl.
Relationale Datenbank	Datenbankkonzept, das Tabellen enthält, welche untereinander verknüpft werden. Die Relationen der Spalten entstehen durch sogenannte Schlüssel.
Tags (InfluxDB)	Zusatzinformationen zu einem Datensatz, nach denen auch gefiltert werden kann. Tags bestimmen auch die Zugehörigkeit eines einzelnen Datensatzes zu einer Zeitreihe.
Zeitreihe	Reihe an Datensätzen, die jeweils bestimmten Zeitpunkten zugeordnet sind. In einer Zeitreihe ist jeder Zeitpunkt einzigartig.
Zeitreihendatenbank	Datenbankkonzept und Untergruppe der relationalen Datenbanken mit optimierten Funktionen für zeitbezogene Daten. Der Zeitstempel eines Datensatzes stellt innerhalb einer Zeitreihe immer einen einzigartigen Primärschlüssel dar.

6 Referenzen

- [1] DB-Engines. Information about relational and NoSQL database management systems. URL: <https://db-engines.com/de/ranking/time+series+dbms>. Zuletzt aufgerufen am 03.02.2021.
- [2] TimescaleDB Blog. TimescaleDB vs. InfluxDB: Purpose built differently for time-series data. 2020.s URL: <https://blog.timescale.com/blog/timescaledb-vs-influxdb-for-time-series-data-timescale-influx-sql-nosql-36489299877/>
- [3] InfluxDB. Key Concepts. URL: https://docs.influxdata.com/influxdb/v1.8/concepts/key_concepts/. Zuletzt aufgerufen am 08.02.2021.
- [4] TimescaleDB Documentation. Data retention. URL: <https://docs.timescale.com/latest/using-timescaledb/data-retention>. Zuletzt aufgerufen am 03.02.2021.
- [5] InfluxDB. Downsampling and data retention. URL: https://docs.influxdata.com/influxdb/v1.7/guides/downsampling_and_retention/. Zuletzt aufgerufen am 03.02.2021.
- [6] InfluxDB. Configure InfluxDB OSS. URL: <https://docs.influxdata.com/influxdb/v1.8/administration/config/>. Zuletzt aufgerufen am 03.02.2021.
- [7] TimescaleDB. Configuring TimescaleDB. URL: <https://docs.timescale.com/latest/getting-started/configuring>. Zuletzt aufgerufen am 03.02.2021.
- [8] InfluxDB Documentation. InfluxDB schema design and data layout. URL: https://docs.influxdata.com/influxdb/v1.8/concepts/schema_and_data_layout/. Zuletzt aufgerufen am 03.02.2021.
- [9] TimescaleDB Documentation. Data model. URL: <https://docs.timescale.com/latest/introduction/data-model>. Zuletzt aufgerufen am 03.02.2021.
- [10] InfluxDB. Cloud. URL: <https://www.influxdata.com/products/influxdb-cloud/>. Zuletzt aufgerufen am 08.02.2021.
- [11] TimescaleDB. Cloud. URL: <https://www.timescale.com/cloud>. Zuletzt aufgerufen am 08.02.2021.
- [12] InfluxDB. OSS guidelines. URL: https://docs.influxdata.com/influxdb/v1.8/guides/hardware_sizing/. Zuletzt aufgerufen am 08.02.2021.

-
- [13] TimescaleDB Blog. Insert Performance.
URL: <https://blog.timescale.com/blog/timescaledb-vs-influxdb-for-time-series-data-timescale-influx-sql-nosql-36489299877/> Zuletzt aufgerufen am 08.02.2021.
- [14] Brussels University. Time Series Databases and InfluxDB. 2017.
URL: https://cs.ulb.ac.be/public/_media/teaching/influxdb_2017.pdf. Zuletzt aufgerufen am 08.02.2021.
- [15] Ghent University. Towards a representative benchmark for time series databases. 2019.
URL: https://lib.ugent.be/fulltxt/RUG01/002/786/062/RUG01-002786062_2019_0001_AC.pdf
- [16] InfluxData Blog. Use Cases.
URL: <https://www.influxdata.com/blog/category/usecase/>. Zuletzt aufgerufen am 08.02.2021.

7 Support und Kontakt

Support

Tel.: +49 911 97282-14
Fax: +49 911 97282-33
E-Mail: support@iba-ag.com

Hinweis



Wenn Sie Support benötigen, dann geben Sie bitte bei Softwareprodukten die Lizenznummer bzw. die CodeMeter-Containernummer (WIBU-Dongle) an. Bei Hardwareprodukten halten Sie bitte ggf. die Seriennummer des Geräts bereit.

Kontakt

Hausanschrift

iba AG
Königswarterstraße 44
90762 Fürth
Deutschland

Tel.: +49 911 97282-0
Fax: +49 911 97282-33
E-Mail: iba@iba-ag.com

Postanschrift

iba AG
Postfach 1828
90708 Fürth

Warenanlieferung, Retouren

iba AG
Gebhardtstraße 10
90762 Fürth

Regional und weltweit

Weitere Kontaktadressen unserer regionalen Niederlassungen oder Vertretungen finden Sie auf unserer Webseite

www.iba-ag.com.