Adaptive Data Compression in Modern Databases: Optimising Analytical Access Patterns for Enhanced Performance - Research Proposal

Piotr Sieminski

October 2023

Introduction

Access patterns are an intrinsic part of the database management systems (DBMS). They refer both to how the users interact with the database as well as how the analytical access patterns (queries) are managed by the DBMS (Galindo et al., 2004, pp.528–538). This research will focus on the execution and management of one single query at a time, as a single user access pattern, and is aiming at enhancing its performance. The idea of this research proposal came from my field of professional work. As I currently hold a position of a Senior Analytics Engineer at an EdTech company, the topic of databases and their potential improvements is immensely absorbing for me.

Research Question

Now, to the research question. I asked myself: ‘How can analytical access patterns in modern databases be effectively optimised for use with adaptive data compression storage in order to achieve enhanced query performance?’

Significance

Why does it matter? Storing and accessing data is widely researched with a multifariousness of general as well as more specialised studies. This study offers a fresh perspective which will address a modern data stack architecture and ever evolving enterprise requirements of accessing large amounts of data with a variety of common database access patterns (Gordon, 2016).

Aims and Objectives

Aim

The main aim of this research is to investigate the potential of enhancing common access patterns to the modern databases leveraging adaptive data compression techniques.

Objectives

Review and Analysis

First and foremost I shall conduct a comprehensive literature review on existing data compression techniques in databases as well as analyse current methodologies employed in modern databases for data compression and storage.

Identification of Access Patterns

Subsequently I shall identify common analytical access patterns in modern databases in order to understand how these patterns influence the efficiency of data retrieval.

Development and Implementation

Next the artefact that will come out as a result of this research would be a framework that enhances performance of the identified access patterns. This artefact should be generalisable and should provide structured improvements that can be applied to numerous (ideally all) query patterns used. Then I would like to implement this framework in a test database environment for empirical analysis.

Performance Evaluation

Next I would need to assess the speed and efficiency of data retrieval and query performance using benchmark queries.

Comparative Analysis

I would like to also perform a comparative analysis that compares the performance of common access patterns with the enhancement framework.

Scalability and Flexibility Assessment

I would also need to look at the scalability and flexibility assessment in order to evaluate how the framework scales with increasing data volumes and varying access patterns as well as I would need to test the flexibility of the framework in adapting to evolving or changing access patterns over time.

Recommendations and Future Directions

The next, a little bit smaller but still important artefact that I would like to come out of this research is a list of best practices and recommendations for database administrators and developers based on research findings.

Last but not least, the future directions, after this research, would be to identify areas for future research, exploring further refinements or applications of the performance optimisation framework.

Key literature

There exists a wide range of literature available for this topic, which has been studied for the past (at least) 45 years - Ziv and Lempel, 1977 - is the oldest publication that is included in this research proposal. When selecting key literature I started with a couple of publications, which were used as an inspiration or basis for modern, to date used developments. The list starts with the Dremel system (Melnik et al., 2011), which was the basis for the Parquet file type (Herberts, 2021). Then the Database Compression (Graefe and Shapiro, 2002) - essential when considering how databases use compression techniques. Last but not least, a book that can be considered a ‘bible’ of dimensional modelling, query structure and overall database design: The Data Warehouse Toolkit written by Kimball and Ross, 2013.

In order to classify the literature for this project I categorised, thematised and extracted keywords for each of the publications (Popenoe et al., 2021). Three main categories of publications emerged: that is Compression Techniques, Database Storage and Design and Access Patterns. The table with classified key literature can be found below.

Theme	Title	Year	Keywords	Category
Access Patterns	Access path selection in a relational database management system	1979	R, access paths, DBMS, IBM	Conference Paper
Access Patterns	What's Really New with NewSQL?	2016	NewSQL, access patterns, DBMS, transactional workloads, analytical workloads	Journal Article
Access Patterns	Real-Time Workload Pattern Analysis for Large-Scale Cloud Databases	2023	cloud database workloads, workload pattern discovery, sql query patterns, access patterns, high-dimensional feature embedding	Conference Paper

Compression Techniques	A universal algorithm for sequential data compression	1977	LZ77, compression, sequential data, algorithm, foundational	Journal Article
Compression Techniques	Database compression	1991	Database, compression, optimization, storage, analytics	Journal Article
Compression Techniques	Integrating Compression and Execution in Column-Oriented Database Systems	2006	Compression, execution, columnar databases, integration, challenges	Conference Paper
Compression Techniques	Super-scalar RAM-CPU cache compression	2006	Compression, RAM, CPU cache, super-scalar, technique	Conference Paper
Compression Techniques	Column-Stores vs. Row-Stores: How Different Are They Really?	2008	Column-stores, row-stores, comparison, database, design	Journal Article
Compression Techniques	Dremel: Interactive Analysis of Web-Scale Datasets	2010	Dremel, columnar storage, Web-Scale, analytics, Google	Journal Article
Compression Techniques	Rethinking SIMD vectorization for in-memory databases	2015	SIMD, vectorization, in-memory, databases, compression	Journal Article
Compression Techniques	SIMD compression and the intersection of sorted integers	2016	SIMD, compression, sorted integers, intersection, algorithms	Journal Article
Database Storage and Design	The implementation and performance of compressed databases	2000	Compression, database performance, adaptive, usage patterns	Journal Article
Database Storage and Design	How to Barter Bits for Chronons: Compression and Bandwidth Trade-Offs for Database Scans	2008	Compression, bandwidth, trade-offs, database scans, chronons	Conference Paper
Database Storage and Design	MonetDB: Two decades of research in column-oriented database architectures	2009	MonetDB, column-oriented, database architectures, research, compression	Journal Article
Database Storage and Design	The Starfish project: Adaptive self-tuning storage systems for the cloud	2011	Starfish, adaptive, self-tuning, storage, cloud	Journal Article
Database Storage and Design & Access Patterns	The Data Warehouse Toolkit	2013	Data warehouse, toolkit, design, star schema, best practices	Book
Database Storage and Design & Access Patterns	Columnar storage and list-based processing for graph database management systems	2021	GDBMS, column-oriented storage, access patterns, list-based query processor\	Conference Paper

Figure 1: Thematic classification table of the key literature.

I will not go through every single article but for the use of the reader of this presentation you can see a ‘Theme’ to the left, ‘Title’ and ‘Year’ of the publication in order to identify them in the references section as well as keywords that describe clearly what each of the publications is mainly about. The last column is ‘Category’ which defines the type of the source of the publication. We can see a mixture of conference papers, journal articles and one, aforementioned, book.

Research Design

First step will be to conduct a full, secondary research (Wickham, 2019). This will greatly inform the subsequent work by providing a clear overview of the existing knowledge (George, 2023), establishing common access patterns that can be analysed as well as providing deep understanding about the adaptive compression techniques. Understanding how the data is stored is imperative when trying to optimise the ways to access it.

Experiment setup

I would like to propose an experimental design of this research. It will involve setting up a controlled, testing environment which will allow to minimise the bias (Song et al., 2013). For this environment, variables such as server load or background processes will be set to constant parameters such that they remain consistent throughout the testing phase.

The next step will be to create numerous instances, with the same base parameters, for each of the databases studied, then split them into treatment and control groups.

It will allow to control for the fact that experiment databases might have some factors (either constant or appearing during runtime) which influence the query performance but are not known to the researcher at the time of executing the tests. Control instances will serve here as benchmarks and will have the same queries run against them at the same time to ensure unbiased results. The memory and compute resources will be aligned between the databases tested in order to even further reduce the possibility of bias (Song et al., 2013). Only databases allowing the use of data partitioning (horizontal or vertical) and adaptive compression techniques will be considered for the study.

Testing

Subsequently, the databases will be subjected to a series of stress tests which will involve various, common access patterns (as previously identified). The study will then progress to exploring the access patterns modifications that are hypothesised to improve performance. Each of the before/after query will be run several times in order to take the sample mean for further statistical testing as well as to reduce the possibility of bias (Song et al., 2013). The metrics monitored will be: runtime, CPU load, GPU load, bytes scanned, disk input/output, number of partitions accessed (accounting for partition size) and query parsing time. Each of these metrics has its own impact on the overall performance as well as costs of executing queries (Batista and Silva, 2013).

Data Analysis Process

Once all of the relevant information is gathered, the statistical tests will be applied in order to establish whether statistically significant improvements (or deterioration)

took place or not. I expect to choose the right statistical test after assessing the structure of the distributions. The normality will be evaluated with a Shapiro-Wilk test (Shapiro and Wilk, 1965). Some of the tests that can likely be chosen for this analysis include: paired (dependent) t-test (Ross and Willson, 2017), Wilcoxon signed-rank test (for not normal distribution) (Rey and Neuhäuser, 2011), repeated measures ANOVA (if there are more than one optimisation applied to a query) (Lix and Keselman, 2018) or a Friedman test (alternative to repeated measures ANOVA which does not assume normality) (Sheldon et al., 1996). Additionally, descriptive statistics and/or p-values will be provided where appropriate in order to provide more context to the results.

Ethical considerations

And now, to ethical considerations.

Data Privacy

The data sets analysed could contain sensitive or personally identifiable information (PII). In order to avoid infringing anyone's rights or privacy, for this research, it is imperative that all of the information stored in the analysed databases is coming from freely and publicly available sources, where (if present) any kind of PII is appropriately anonymised.

Intellectual Property

If any of the analysed databases, data sets or specific algorithms is considered proprietary, relevant permissions and/or licences will be acquired in order to obtain the right to use, modify or analyse them.

Transparency in reporting

It is crucial that the report of the performance is given in full detail with appropriate analysis and descriptions (where applicable). Any and all assumptions as well as specific computing specifications should be included in the report not only to allow for reproducibility of the results but also to ensure that no future researchers are misled.

Environmental impact

Last but not least, the environmental impact.

Large or repetitive query execution as well as large data sets storage can consume significant computational power and storage, causing increased carbon footprint (Abeydeera et al., 2019). Therefore using green energy and/or computing sources will be chosen where possible.

Risk assessment

Risk	Mitigation
Data Breach	Public data, no PII
Data Corruption	Spot checks, original copies
Dialect specific optimisations	Exclusion from general findings
Dependence on proprietary tools	Open source tools, licence
Outdated technology	Latest developments used

Figure 2: Risk assessment analysis table.

On this slide you can see the Figure number 2, which is the risk analysis table.

First one is the ‘Data Breach’. Infringement of someone’s privacy, risk of PII information leakage can happen and in order to prevent that from happening the usage of publicly with no PII will be commissioned for this research. Next is ‘Data Corruption’. Adaptive compression and then decompression techniques could cause unintentional data corruption. Because of that, original copies/sources of the data will be stored separately. Data will also be spot validated. ‘Dialect specific optimisations’: it can happen that some dialects will have identified improvements that are specific to this particular dialect in that case this finding will be excluded from the general finding sections and it will be duly noted on the report that there is an improvement found but it applies to this and this only specific SQL dialect. ‘Dependence on proprietary tools’: relying heavily on proprietary software for research can cause licensing or reproducibility issues. So wherever possible, open source tools will be used and if needed - proper licence will be obtained. Last but not least: ‘Outdated technology’: The rapid pace of technological advancement can cause this research to become outdated quickly and so this research will use the latest developments in the field as well as consider future directions.

Timeline of proposed activities

Activity	Start	End
Project Set Up/Supervisor consultation	13/11/2023	11/12/2023
Secondary research	11/12/2023	05/02/2024
Database Selection	05/02/2024	12/02/2024
Data Partitioning	12/02/2024	19/02/2024
Stress Testing	19/02/2024	11/03/2024
Access Patterns Modifications	11/03/2024	08/04/2024
Data Analysis	08/04/2024	22/04/2024

Final Analysis and Reporting	22/04/2024	06/05/2024
Feedback gathering	06/05/2024	20/05/2024
Conclusion and Documentation	20/05/2024	03/06/2024

Figure 3: Proposed activities timeline - table.

The timeline of the proposed activities. You can see the figure number three, which describes the activity as well as starting and ending point. In order to digest this information better I created a Gantt chart.

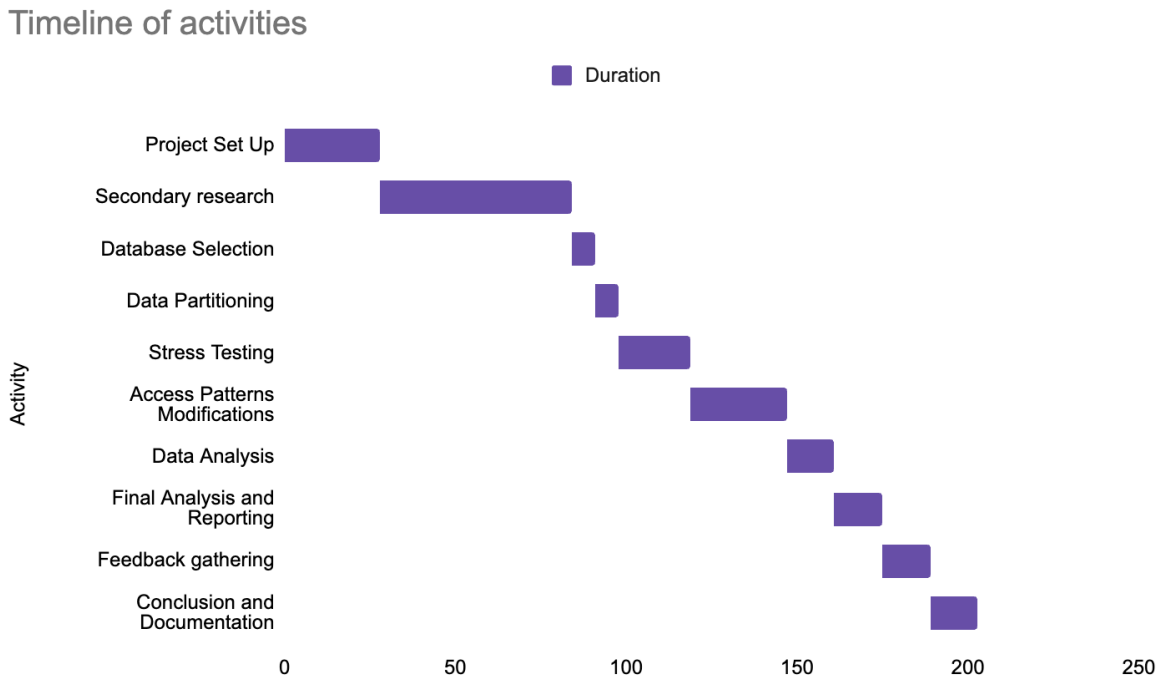

Figure 4: Gantt chart visualising the duration of each of the proposed activities in days.

You can see that the ‘Secondary Research’ is taking up the most space as it is, as aforementioned, imperative in order to conduct a fully informed primary research. Then ‘Stress Testing’ and ‘Access Patterns Modifications’: other two activities taking up quite a lot of time where I will, so to say: ‘get my hands dirty’ in testing and experimenting in all of the databases and I expect to have a full, valid set of results that I can later on analyse, prepare for feedback, share this with supervisors, maybe

peers and then glean all of the information together so that I can reach a conclusion, prepare full documentation and submit the work.

Conclusion

This presentation proposes to study databases' common analytical access patterns improvements in an experimental way. It aims to contribute to the field by examining the newest technology and developments, using adaptive data storage techniques as a prerequisite for a database to take part in the experiment. I believe that all ethical considerations and risks are identified and a proper plan is set in place in order to ensure that the research will adhere to the ethics code (Stahl et al., 2016) and that no risk can jeopardise its successful completion.

Reference list

Abadi, D.J., Madden, S. and Ferreira, M. (2006). Integrating compression and execution in column-oriented database systems. *International Conference on Management of Data*. [online] doi:https://doi.org/10.1145/1142473.1142548.

Abadi, D.J., Madden, S.R. and Hachem, N. (2008). Column-stores vs. row-stores. *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pp.967–980. doi:https://doi.org/10.1145/1376616.1376712.

Batista, E. and Silva, M.B. (2013). Influential Parameters to the Database Performance — A Study by Means of Design of Experiments (DoE). In: M.B. Silva, ed., *Design of Experiments - Applications*. [online] IntechOpen. doi:https://doi.org/10.5772/56546.

Burns, R.K. and Koh, Y. (1996). A comparison of multilevel structured query language (SQL) implementations. In: *Proceedings 12th Annual Computer Security Applications Conference*. [online] 12th Annual Computer Security Applications Conference. Available at: 10.1109/CSAC.1996.569698 [Accessed 11 Oct. 2023].

Galindo, F., Makoto Takizawa and R Traunmüller (2004). *Database and expert systems applications : 15th international conference, DEXA 2004, Zaragoza, Spain, August 30-September 3, 2004 : proceedings*. Berlin ; London: Springer, pp.528–538.

George, T. (2023). *What is Secondary Research? | Definition, Types, & Examples*. [online] Scribbr. Available at: https://www.scribbr.com/methodology/secondary-research/.

Gordon, R. (2016). *Evolution in Databases to Meet Changing Business Needs*. [online] Mainline. Available at: https://mainline.com/evolution-databases-meet-changing-business-needs/ [Accessed 13 Oct. 2023].

Graefe, G. and Shapiro, L.P. (2002). Data compression and database performance. *CiteSeer X (The Pennsylvania State University)*. doi:https://doi.org/10.1109/soac.1991.143840.

Gupta, P., Mhedhbi, A. and Salihoglu, S. (2021). Columnar storage and list-based processing for graph database management systems. *Proceedings of the VLDB Endowment*, 14(11), pp.2491–2504. doi:https://doi.org/10.14778/3476249.3476297.

Herberts, M. (2021). *Demystifying the use of the Parquet file format for time series*. [online] SenX. Available at: https://blog.senx.io/demystifying-the-use-of-the-parquet-file-format-for-time-series/ [Accessed 11 Oct. 2023].

Herodotou, H. (2011). Starfish: A Self-tuning System for Big Data Analytics. In: *CIDR 2011, Fifth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 9-12, 2011, Online Proceedings*. [online] Available at: https://www.researchgate.net/publication/220988112_Starfish_A_Self-tuning_System_for_Big_Data_Analytics [Accessed 11 Oct. 2023].

Holloway, A.L., Raman, V., Swart, G. and DeWitt, D.J. (2007). How to barter bits for chronons: compression and bandwidth trade offs for database scans. In: *SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data*. [online] pp.389–400. Available at: https://doi.org/10.1145/1247480.1247525 [Accessed 11 Oct. 2023].

Idreos, S., Groffen, F., Nes, N., Manegold, S., Mullender, S. and Kersten, M. (2012). MonetDB: Two Decades of Research in Column-oriented Database Architectures. *Bulletin of the Technical Committee on Data Engineering*, [online] 35. Available at: https://hdl.handle.net/11245/1.380005 [Accessed 11 Oct. 2023].

Kimball, R. and Ross, M. (2013). *The data warehouse toolkit : the definitive guide to dimensional modeling*. 3rd ed. John Wiley & Sons, Inc., pp.443–496.

Lemire, D., Boytsov, L. and Kurz, N. (2015). SIMD compression and the intersection of sorted integers. *Software: Practice and Experience*, 46(6), pp.723–749. doi:https://doi.org/10.1002/spe.2326.

Lix, L.M. and Keselman, H.J. (2018). Analysis of Variance. *The Reviewer's Guide to Quantitative Methods in the Social Sciences*, pp.15–28. doi:https://doi.org/10.4324/9781315755649-2.

Melnik, S., Gubarev, A., Long, J.J., Romer, G., Shivakumar, S., Tolton, M. and Vassilakis, T. (2011). Dremel. *Communications of the ACM*, 54(6), pp.114–123. doi:https://doi.org/10.1145/1953122.1953148.

Pavlo, A. and Aslett, M. (2016). What's Really New with NewSQL? *ACM SIGMOD Record*, [online] 45(2), pp.45–55. doi:https://doi.org/10.1145/3003665.3003674.

Polychroniou, O., Raghavan, A. and Ross, K.A. (2015). Rethinking SIMD Vectorization for In-Memory Databases. In: *SIGMOD '15: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. [online] pp.1493–1508. Available at: https://doi.org/10.1145/2723372.2747645 [Accessed 11 Oct. 2023].

Popenoe, R., Langius-Eklöf, A., Stenwall, E. and Jervaeus, A. (2021). A practical guide to data analysis in general literature reviews. *Nordic Journal of Nursing Research*, [online] 41(4), pp.175–186. doi:https://doi.org/10.1177/2057158521991949.

Ramaiah, R. (2022). *Parquet file format, its Dremel connection and how it gets along with Google ProtoBuf*. [online] Medium. Available at: https://medium.com/@ramu.ramaiah/parquet-file-format-its-dremel-connection-and-how-it-gets-along-with-google-protobuf-7b1f221ddcdf [Accessed 11 Oct. 2023].

Rey, D. and Neuhäuser, M. (2011). Wilcoxon-Signed-Rank Test. *International Encyclopedia of Statistical Science*, [online] pp.1658–1659. doi:https://doi.org/10.1007/978-3-642-04898-2_616.

Ross, A. and Willson, V.L. (2017). *Basic and Advanced Statistical Tests*. [online] Rotterdam Sensepublishers. Available at: https://doi.org/10.1007/978-94-6351-086-8_4 [Accessed 11 Oct. 2023].

Selinger, P.G., Atrahan, M.M., Chamberlin, D.D., Lorie, R.A. and Price, T.G. (1979). Access path selection in a relational database management system. In: *SIGMOD '79: Proceedings of the 1979 ACM SIGMOD international conference on Management of data*. [online] Available at: https://doi.org/10.1145/582095.582099 [Accessed 11 Oct. 2023].

Shapiro, S.S. and Wilk, M.B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, [online] 52(3-4), pp.591–611. doi:https://doi.org/10.1093/biomet/52.3-4.591.

Sheldon, M.R., Fillyaw, M.J. and Thompson, W.D. (1996). The use and interpretation of the Friedman test in the analysis of ordinal-scale data in repeated measures designs. *Physiotherapy Research International*, 1(4), pp.221–228. doi:https://doi.org/10.1002/pri.66.

Song, F., Hooper, L. and Loke, Y. (2013). Publication bias: what is it? How do we measure it? How do we avoid it? *Open Access Journal of Clinical Trials*, p.71. doi:https://doi.org/10.2147/oajct.s34419.

Stahl, B.C., Timmermans, J. and Mittelstadt, B.D. (2016). The Ethics of Computing. *ACM Computing Surveys*, 48(4), pp.1–38. doi:https://doi.org/10.1145/2871196.

Udara Willhelm Abeydeera, L.H., Wadu Mesthrige, J. and Samarasinghalage, T.I. (2019). Global Research on Carbon Emissions: A Scientometric Review. *Sustainability*, 11(14), p.3972. doi:https://doi.org/10.3390/su11143972.

Wang, J., Li, T., Wang, A., Liu, X., Lu, C., Chen, J., Liu, J., Wu, J., Li, F. and Gao, Y. (2023). Real-Time Workload Pattern Analysis for Large-Scale Cloud Databases. *Proceedings of the VLDB Endowment*, 16(12), pp.3689–3701. doi:https://doi.org/10.14778/3611540.3611557.

Westmann, T., Kossmann, D., Helmer, S. and Moerkotte, G. (2000). The implementation and performance of compressed databases. *ACM SIGMOD Record*, 29(3), pp.55–67. doi:https://doi.org/10.1145/362084.362137.

Wickham, R. (2019). Secondary Analysis Research. *Journal of the Advanced Practitioner in Oncology*, [online] 10(4), pp.395–400. doi:https://doi.org/10.6004/jadpro.2019.10.4.7.

Ziv, J. and Lempel, A. (1977). A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 23(3), pp.337–343. doi:https://doi.org/10.1109/tit.1977.1055714.

Zukowski, M., Heman, S., Nes, N. and Boncz, P. (2006). Super-Scalar RAM-CPU Cache Compression. In: *ICDE ’06: Proceedings of the 22nd International Conference on Data Engineering*. [online] Available at: https://doi.org/10.1109/ICDE.2006.150 [Accessed 11 Oct. 2023].

Icons’ credits section

This section is created in line with the requested referencing style of the website from which the icons were downloaded (https://www.flaticon.com/).

DBMS icon: Dbms icons created by Vectors Tank - Flaticon

Access icon: Access control icons created by Uniconlabs - Flaticon

Database icon: Big data icons created by The Chohans Brand - Flaticon

Binoculars icon: Binoculars icons created by Freepik - Flaticon

Technique icon: Technique icons created by amoghdesign - Flaticon

Database icons: Database icons created by phatplus - Flaticon

Access icon: Access control icons created by Uniconlabs - Flaticon

Testing environment icon: Customer experience icons created by Rmpp - Flaticon

Constant parameters icon: Load balancer icons created by small.smiles - Flaticon

DB instances icon: Replicate icons created by Vectors Tank - Flaticon

Stress icon: Stress icons created by Freepik - Flaticon

Normal distribution icon: Normal Distribution icons created by orvipixel - Flaticon

Statistical test icon: Decision icons created by judanna - Flaticon

Descriptive metrics icon: Metrics icons created by Uniconlabs - Flaticon

Data privacy icon: Privacy policy icons created by Flat Icons - Flaticon

Intellectual property icon: Intellectual icons created by Eucalyp - Flaticon

Environmental impact icon: Transparency icons created by Wichai.wi - Flaticon

Transparency in reporting icon: Esg icons created by kmg design - Flaticon