Journal of Data, Information and Management (2026) 8:71–96
https://doi.org/10.1007/s42488-026-00162-x

RESEARCH

Comprehensive insights into bitemporal databases: a PRISMA-guided systematic literature review

Gokul Neelamegam[1] · Jagdev Bhogal[1] · Parnia Samimi[1] · Omer Ozturkoglu[2]

Received: 17 February 2026 / Accepted: 25 April 2026 / Published online: 11 May 2026

Abstract
Conventional databases support either retroactive or proactive changes, but not both simultaneously, resulting in the loss of valuable historical data. Bitemporal databases resolve this problem by supporting two temporal dimensions simultaneously: valid time and transaction time. This approach enhances data integrity and traceability in real-world applications. Although numerous studies have examined bitemporal databases, a comprehensive review of the field is still lacking. Therefore, we conducted a hybrid systematic literature review following the PRISMA framework, supplemented by bibliometric and scientometric analyses, covering publications from January 1995 to January 2026. Four main databases were chosen: IEEE, Scopus, ProQuest, and Web of Science. Subsequently, 54 papers were analysed, providing unique insights into bitemporal data modelling, querying, and applications. Key gaps identified include the need for hybrid Data Base Management System (DBMS) infrastructures, limited empirical validation on industrial-scale datasets, insufficient cloud-native temporal optimisation strategies, and underdeveloped integration of bitemporal data structures with AI-driven predictive analytics. Finally, this study summarises current progress in bitemporal database research, highlights key implementation challenges, and emphasises its strong potential to improve data accuracy, traceability, and temporal consistency across diverse application areas.

Keywords Bitemporal data · Systematic literature review · PRISMA · Bibliometric analysis · Data integrity

1 Introduction

Temporal studies of databases have developed alongside data modelling approaches. Early discussions on the temporal aspects of data systems date back to 1956, well before the relational model and the emergence of early database management systems. However, significant research momentum only gained traction after the relational model was formalised, with several proposals made to represent time in databases (Kvet et al. 2014a). Time is a crucial attribute of information. Neither traditional relational databases (RDB) nor object-oriented databases (OODB) explicitly handle temporal data. Both lack integrated capabilities for recording and managing temporal information. As the latest advancement in database and information technology, the need for temporal information processing is becoming increasingly vital in many application systems (Tang et al. 2004).

A relational database stores data in tables, where each table consists of attributes and tuples, whereas a temporal database stores data in time instances, formed by timestamping ordinary data. Timestamping associated with a tuple is called tuple timestamping; when associated with each attribute, it is called attribute timestamping. Temporal data access can involve several aspects, including temporal data models, query languages, and access methods (Salzberg and

✉ Gokul Neelamegam
gokul.neelamegam@mail.bcu.ac.uk

Jagdev Bhogal
jagdev.bhogal@bcu.ac.uk

Parnia Samimi
parnia.samimi@bcu.ac.uk

Omer Ozturkoglu
omer.ozturkoglu@bcu.ac.uk

[1] Department of Computer Science, School of Architecture, Built Environment, Computing, and Engineering, Birmingham City University, Birmingham, UK

[2] Department of Management, Business and Marketing, Business School, Birmingham City University, Birmingham, UK

Tsotras 1999). Temporal information processing has become a key technology in the new generation of databases and information systems. Temporal database systems track data evolution by associating timestamps with each data item. All changes are recorded, and past states of the database can be retrieved. Two different timestamps can be used: the transaction time, which corresponds to the moment the information is introduced into the database, and the valid time, which indicates when that information is valid in the real world (Johnston 2014). However, conventional databases assume that all stored facts are "true," based on a two-valued logic in which data are present, absent, or false, and any update replaces all previous information. However, bitemporal databases overcome this limitation by providing valid time and transaction time, allowing the system to track whether facts are true in real life or in the database. This method preserves historical data through controlled insertions, deletions, and updates that adjust valid periods or produce new records rather than overwriting old ones. Thus, bitemporal tables combine valid time with transaction time, enabling the management of changing facts while tracking when those facts were true and allowing more flexible historical queries (Johnston and Weis 2010).

In an era of rapidly evolving data management needs, understanding bitemporal databases is essential for capturing, querying, and analysing information across multiple dimensions of time (Kaufmann et al. 2014). Bitemporal time in the database extends beyond mere fact storage. It records both the time period in which facts were valid and the time at which the database captured that fact, such as time-travelling. A bitemporal database introduces a second time dimension to the data, referred to as the business dimension. By integrating both valid time and transaction time dimensions, bitemporal databases enable organisations to maintain accurate historical records, support informed decision-making, and ensure temporal consistency in complex systems (Tansel et al. 2025).

Although bitemporal data support dual time dimensions, there is a gap between the bitemporal support in research work and that required in real-world applications. This occurred because very few works utilised the full potential of bitemporal database design (Chau and Chittayasothorn 2021). This resulted in slow adoption in commercial applications. This has recently begun to change, as many organisations are capturing and querying past databases to understand underlying real-world events. After the introduction of SQL:2011 standard, the popular commercial language for the relational model has been improved with added features such as the ability to create and manipulate system-versioned temporal tables (Tang et al. 2004).

Moreover, recently, commercial DBMSs have supported bitemporal features. In this context, the use of these features has gained prominence for identifying past data patterns, making predictions and reducing storage ambiguity in model training and evaluations (Bliujute et al. 2000). Furthermore, as bitemporal features continue to expand, it becomes crucial to evaluate and compare how existing models support dual dimensions, such as valid time and transaction time.

This study aims to bridge the gap between theoretical advances and the practical applications of bitemporal databases, providing insights to guide researchers and practitioners in leveraging temporal data for robust, reliable information management. The overall structure of this paper follows established systematic literature review reporting practices, adapted to a hybrid methodology integrating systematic review procedures with bibliometric and scientometric analyses (Asghar et al. 2019). The remainder of this paper is organised as follows. Section II presents the background, related work, and motivation for the study. Section III describes the research methodology adopted for this study. Section IV covers the results and discussion of the SLR, with topics structured to address the research questions, including research gaps, limitations, and the future direction of bitemporal databases. Finally, the conclusion.

2 Background

A bitemporal database is a relational structure that maintains two temporal dimensions: system-supported valid time and transaction time. Regarding valid time, it refers to the time at which the fact is collected, which can span the past, present, and future, as well as the time at which the fact is true in the modelled reality. By definition, all facts have a valid time. However, the valid time might not always be explicitly recorded within the database (B¨ohlen et al. 2018a). In some cases, multiple valid times can be associated with a single fact, each represented by a separate database tuple. Transaction time refers to the moment when a fact is stored in the database (Cappelli et al. 1995). Unlike valid time, transaction time may be associated with any database entity, not only with facts. Transaction time may be implemented using transaction commit time, system-generated, or supplied (Davies et al. 1995). Also, transaction time is particularly well-behaved due to its special semantics and may be automatically supplied by the Database Management System (DBMS). This explains why most temporal database research has focused on improvements in valid-time and transaction-time management (Kvet et al. 2014b).

From the late 1990s onward, valid time versus transaction time was a distinguishing feature in temporal database research and the basis for formal language work and system designs. Following that, Mareco and Bertossi (1999) examine the data response in terms of changes and schema design, handling them using an object-oriented approach. They aim to support long-term system evolution, reuse, and traceability in complex environments by preserving earlier states and allowing changes in the controller's behaviour over time. Temporal Structured Query Language (TSQL2), for example, sought to provide a unified semantic construct for valid time, transaction time, and their combinations. However, TSQL2 itself did not become an International Organisation for Standardisation (ISO) standard but influenced many later standards and implementations and remains an important reference for bitemporal modelling (Dyreson 1995). Then, the concept of adding time to databases becomes a necessary feature of temporal databases. In conventional databases, information is lost over time because data is usually associated with the current time. The main goal of bitemporal databases is to preserve data that exists at different points in time. To overcome that, a database would typically keep a "snapshot" of reality. This is called a snapshot database. For many use cases, this just means that historical values are overwritten when new information arrives, but historical data tracking is necessary for data analysis, auditing, and legal documentation (Bohlen et al. 2000). This requirement highlighted the need for bitemporal databases, but it took until the end of 2011 to finalise the SQL standard and introduce SQL:2011, which represented a significant step in temporal data management. Instead of introducing new data types, it defines time periods in existing date-time columns (Kulkarni and Michels 2012).

The SQL standardisation of bitemporal support reflects a growing need to handle data across multiple temporal dimensions without major changes to database structures. Dual timeline management, which includes valid time and transaction time, has become indispensable in banking risk assessment, inventory control, healthcare, legal auditing, and supply chain management. They give organisations the ability to run complex analyses, keep full historical records, and meet regulatory requirements more precisely (Du and Zhu 2024). In today's enterprise systems, bitemporal databases are of great importance for data integrity and operational resilience. However, despite decades of research, adoption and implementation of the standard are progressing steadily. As of the 2018 survey, IBM DB2, Oracle, and Teradata offer bitemporal support, while other systems, such as PostgreSQL, have room for improvement. In practice, bitemporal can be achieved either by modifying existing relational schemas or by using extensions and helper functions in object-relational and NoSQL tools (B¨ohlen et al. 2018b).

2.1 Motivation

The motivation for this review stems from the increasing complexity of temporal data management and the need for precise historical accountability, making bitemporal databases a central area of inquiry in data management research. While considerable progress has been made, terminology, modelling, and system capabilities remain inconsistent across the literature and commercial implementations. Therefore, a systematic literature review is necessary to synthesise existing knowledge, methodological advances, and the alignment between theoretical frameworks and practice. It will review unresolved issues, highlight emerging research themes, and lay the foundation for future developments in bitemporal data modelling and systems design.

2.2 Related work

To begin this study, we searched the following academic databases and search engines, such as Google Scholar, Scopus, IEEE, Web of Science, ProQuest, and PubMed, with the search strings.

("Systematic study" OR "Literature review" OR "SLR" OR "Systematic mapping") AND ("bitemporal database" OR "bitemporal data model" OR "Dual Dimension" OR "Bitemporal").

We found no systematic literature review or mapping study that addresses the development and implementation challenges, as well as cross-domain applications, of bitemporal databases. However, historical data remains a critical area in temporal database management, and recent research on bitemporal data models has attracted increasing attention in areas such as fraud detection, regulatory compliance, and digital forensics. In the following, we first describe all relevant studies that are directly related to the quality of bitemporal database applications.

Jensen and Snodgrass's (1996) early work presents a foundational study on the semantics of time-varying data in databases, with a particular focus on bitemporal information. The paper clearly defined a valid time dimension that describes when a fact holds in the real world, distinct from transaction time, which captures when the fact is stored in the databases. The authors introduce the Bitemporal Conceptual Data Model (BCDM), a conceptual model that separates temporal meaning from physical representation and preserves information content through the principle of snapshot equivalence, enabling different temporal models to be compared consistently. Overall, the work serves as a key theoretical reference for bitemporal databases by clearly defining temporal meaning independently of implementation details.

Snodgrass (1999) presents a comprehensive study on the bitemporal database, formalising the valid-time and transaction-time dimensions. One key aspect of the research involves introducing TSQL2 (Temporal Structure Query Language) queries within the SQL framework, which reduces the complexity of managing time-based data by unifying temporal vocabulary and semantics. This later resulted in a standardised temporal concept that influenced the development of bitemporal extensions in the ISO SQL:2011 standard. Furthermore, it involves investigating temporal anomalies and conflicts, such as database inconsistencies that can cause overlapping time intervals and retroactive issues corrections. Thus, it provided the groundwork for further research on bitemporal indexing, optimization, and query processing (Jensen et al. 1994).

While Jensen and Snodgrass established the theoretical foundations of bitemporal semantics, Kumar et al. (1998) shifted the focus toward practical implementation, addressing storage complexity and indexing strategies that bridge the gap between conceptual models and real-world system performance.

Kumar et al. (1998) systematically study the challenges of managing bitemporal data, particularly focusing on storage complexity and retrieval methods. They mainly analyse and propose access techniques and indexing strategies to enhance performance, bridging the gap between theoretical models and the implementation of bitemporal databases. Furthermore, they also discussed how bitemporal indexing could improve query optimisation and historical data retrieval, providing a framework that influenced bitemporal database designs. Overall, they believe that their contribution marked a shift from conceptual temporal modelling towards high-performance solutions for managing bitemporal systems.

Johnston's (2014) work provides the most comprehensive treatments of bitemporal data, both in theory and in practice. Furthermore, it explains how bitemporal dimensions can serve as robust alternatives to Kimball's slowly changing dimensions in star schemas. However, while previous work mainly addresses data quality, validation processes, and overall bitemporal challenges, this publication offers a far more specialised and comprehensive examination of bitemporal data, emphasising the need for both conceptual understanding and technical competence in developing and deploying bitemporal systems. In addition, it discusses relational theory, ontology, mathematics, and semantics to highlight why conventional relational database systems are limited in their ability to handle bitemporal functionality. Finally, it explores the business value of bitemporal databases. Overall, it summarises academic research on practical database design and represents a significant step toward a fuller understanding of bitemporal databases.

In contrast to the system-level focus of Kumar et al. (1998), Johnston (2014) takes a broader perspective by examining the business value and relational theory of bitemporal databases, highlighting that the challenges extend beyond technical implementation to organisational adoption and data governance.

Chau and Chittayasothorn (2021) point out and summarise the issues faced by bitemporal research capabilities and their integration into real-world database design, and propose a method to incorporate temporal features in healthcare domains with inherently time-sensitive data. However, the methods lack temporal consistency, treating valid and transaction time as conventional attributes. To address this, they use topological sorting to ensure a correct sequence of mappings, where each Enhanced Entity-Relationship (EER) construct is mapped to a system-versions table with dual time dimensions, adhering to the SQL closed-open period schema. Furthermore, their approach aims to reduce the complexity of integrating bitemporal databases into temporal-rich domains, such as medical databases.

Tansel et al. (2025) consider bitemporal functionality for their research by designing an innovative extension of the Resource Description Framework (RDF) to seamlessly incorporate bitemporal database features into the semantic web environment. However, this approach differs fundamentally from standard RDF, which is limited to static binary relationships in the form of subject-predicate-object triples. Their approach is intended to simplify semantics, enhance flexibility by separating time from core attribute values, and ensure compatibility with the theory of bitemporal databases. Furthermore, they believe that their BiTRDF model allows for navigation and manipulation of the temporal dimension without altering the underlying RDF structure.

While Chau and Chittayasothorn (2021) focus on domain-specific integration of bitemporal features in healthcare, Tansel et al. (2025) take a fundamentally different approach by extending the semantic web framework with dual temporal dimensions, demonstrating that bitemporal principles are applicable beyond traditional relational database environments.

To sum up, the existing studies show that bitemporal database applications provide significant opportunities to manage time-dependent data, enabling organisations to enhance promotional models (Wondoh et al. 2016). Additionally, research explored the applicability of bitemporal data management across various domains, including weather forecasting, the supply chain, and the financial sector, where accurate historical and transactional data are critical for reliable prediction and decision-making (Sedighi et al. 2020). Despite the gradual growth of the literature on bitemporal databases and their technical implementation, current studies provide a limited synthesis of how bitemporal concepts are communicated and integrated into practical, real-world systems. This gap presents a significant opportunity for further exploration. To this end, we identify existing research on bitemporal database applications and provide a structural understanding of their characteristics, usage patterns, and limitations.

3 Research method

A systematic literature review (SLR) is a structured and well-organised research for identifying, assessing, and synthesising existing research on a specific topic (Hilmi et al. 2023). It follows a precise and transparent process to ensure that all relevant studies are included. By following a systematic and well-defined procedure, this type of study offers many advantages. First, systematic mappings enable a comprehensive overview of the current state of research in a particular critical area. Second, it facilitates the identification of gaps in the literature and provides evidence to guide further research, thereby preventing redundant efforts (Adnan et al. 2021). Systematic mappings provide a high-level analysis of all available studies within a specific domain, offering insights into broad research questions about the current state of the literature.

In this paper, a systematic methodology is employed based on the PRISMA framework, which provides a qualitative synthesis of the study, while a bibliometric and scientometric analysis across four major databases, including IEEE Xplore, Scopus, ProQuest, and Web of Science, provides a quantitative evaluation.

An SLR is typically conducted in three main phases. The planning phase defines the research questions, outlines the search strategy, and establishes criteria for selecting relevant studies. The execution phase identifies and selects the required studies based on the results of the previous phase. The reporting phase analyses the selected studies and identifies limitations and future research directions, as shown in Fig. 1.

4 Research questions

In the initial stage of the SLR, research questions were identified to address the study's objectives. Then, the relevant data search sources, inclusion and exclusion criteria, and quality assessment criteria have been presented in this planning phase.

Research questions typically guide us in conducting a comprehensive study and achieving specific research objectives. We adopted the Goal-Question-Metric (GQM) approach (Farina et al. 2022). Applying the Goal-Question-Metric method helps develop a measurement system for defined problems and set rules for interpreting measurement data. To set our research toward a specific direction, we figured out the goal of our SLR based on the following GQM components, such as purpose, issue, objective and viewpoints, which are described in Table 1.

In Table 2, The research questions are listed, along with their corresponding primary objectives.

4.1 Data source selection

A crucial step in developing any SLR is formulating a search strategy. Given the number of papers on the bitemporal topic, we decided to use multiple databases to cover

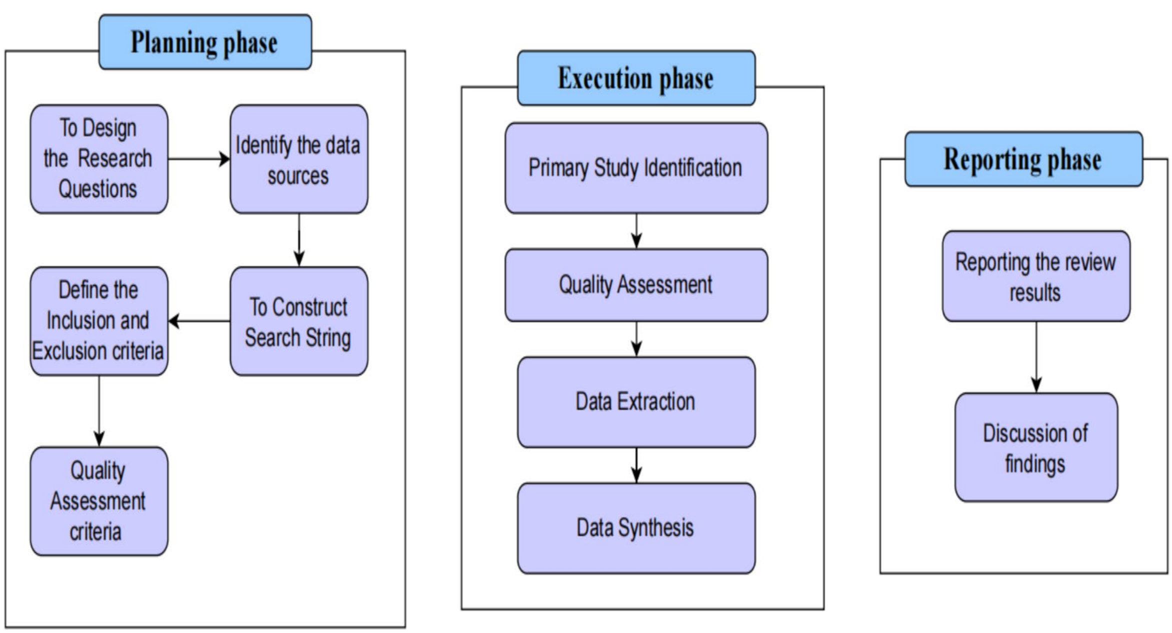


Fig. 1 Phases of systematic literature review

Table 1 GQM component description

GQM component	Description
Purpose	To understand bitemporal database models, applications, and research trends
Issue	To address the fragmentation in existing studies and the lack of systematic consolidation of knowledge on bitemporal databases
Objective	To systematically review and synthesise research on bitemporal databases
Viewpoint	To map knowledge, identify trends, and highlight gaps for future research from a researcher's perspective

Table 2 Research questions and objectives

ID	Research question	Objectives
RQ1	What are the publication patterns, leading contributors, and emerging research themes in the field of bitemporal databases?	To analyse publication patterns in the field of bitemporal databases, including trends over time, key research papers, and prominent authors To explore emerging research themes
RQ2	How do bitemporal database models and implementations compare in terms of data representation, query capabilities, and temporal semantics?	To analyse different bitemporal database models and implementations for temporal data representation To evaluate the query capabilities and temporal semantics of existing bitemporal models To explore how time-based queries are processed and how temporal information is read and managed
RQ3	What are the emerging use cases of bitemporal databases across different domains?	To examine which of these use cases have been implemented in real-world systems and assess their practical relevance
RQ4	What are the future research directions and open challenges in bitemporal database design, implementation, and query processing?	To provide a roadmap for researchers to focus on emerging challenges, applications, and innovations in bitemporal databases

all potential publications. The databases we used include Scopus, IEEE Xplore, ProQuest and Web of Science. These databases were selected due to their comprehensive coverage of both technical and interdisciplinary publications. The data sources mentioned in this review were selected for their high-impact, high-quality publications. The summary of studies for each selected data source link is listed in Table 3, along with the number of papers in each database.

4.2 Search strategy

We used keywords related to bitemporal databases to identify relevant literature. The selected keywords were sufficiently general to ensure we would not miss any important papers.

Table 3 Identification of studies from each database

Data sources	Identification	Link
IEEE Xplore	49	https://ieeexplore.ieee.org
Scopus	102	https://www.scopus.com
ProQuest	23	https://www.proquest.com
Web of Science	74	https://www.webofscience.com

We aim to identify papers that align with the research questions, specifically regarding strengths, limitations, and use cases. We applied the Title, Abstract, and Keywords fields of the articles commonly denoted as TITLE-ABS-KEY to ensure comprehensive coverage of relevant studies.

Additionally, the asterisk (*) wildcard symbol was used to capture multiple variations of root terms, thereby broadening the scope of the search and including all possible word forms related to the topic. We used Boolean operators to do so, in accordance with the best practices of our discipline.

After that, we searched the originally selected databases using these Search Queries.

The search string for this study is as follows

TITLE-ABS-KEY("bitemporal data" OR "bi-temporal data" OR"bitemporal database" OR "bi-temporal database" OR "bi temporal database*" OR "bi-temporal model*" OR "bitemporal model*" OR "bitemporal model*").

AND

("strength" OR "advantage" OR "benefit*" OR "weakness*" OR "limitation*" OR "challenge*" OR "drawback*" OR "issue*" OR "problem*" OR "constraint*" OR "use case*" OR "application*" OR "scenario*" OR "case stud*" OR "implementation*").

The search string was organised into two logical parts: the first focuses on the core bitemporal topic, while the second filters for studies that examine use cases, implementation scenarios, and evaluation outcomes directly linked to the research questions established through the Goal-Question-Metric (GQM) framework (Farina et al. 2022). This two-part Boolean structure adheres to recognised SLR search design practices to balance recall and precision. Terms such as "use case*", "application*", and "implementation*" are descriptive rather than evaluative and are commonly included in technical database papers, reducing the risk of omitting relevant technical studies. Although terms such as "drawback", "issue", and "problem" may seem evaluative, they are frequently used in technical database literature to describe system limitations and design constraints, and their inclusion helps ensure that studies reporting implementation challenges, especially those relevant to RQ3 and RQ4, are not unintentionally excluded (Hilmi et al. 2023).

Grey literature not published in the four selected databases, such as technical reports, was excluded. Although these can be valuable sources of information, they may not undergo rigorous peer review. To ensure the quality and validity of the studies included in our SLR, we avoided grey literature. We utilised the Zotero tool (Kaur and Dhindsa 2016) to manage and organise the selected articles (Moumen et al. 2020). This tool improved efficiency in sorting and removing duplicate studies across different scientific journals in this review.

4.3 Inclusion and exclusion criteria

The determination of inclusion and exclusion criteria is important for identifying the most relevant studies to address the research questions.

The inclusion criteria for this study comprised studies published between January 1995 and January 2026, written in English, and published as peer-reviewed journal articles, conference proceedings, or book chapters with relevant keywords. The selected time period was chosen to reflect the transition from early conceptual temporal database research to more formalised bitemporal models and standards, particularly following the introduction of TSQL2, which laid the foundation for bitemporal database design by enabling the representation of both valid-time and transaction-time dimensions (Dyreson 1995), while earlier foundational works are acknowledged in the background section.

Furthermore, only studies meeting the quality assessment criteria were deemed eligible for inclusion. The exclusion criteria included studies published outside the designated time frame, in languages other than English, or classified as grey literature, such as technical reports, magazines, opinion pieces, or editorials. Publications that were not peer-reviewed and duplicate records were also excluded. The Inclusion and Exclusion Criteria are formulated as (IC) and (EC) to ensure a systematic and coherent way to satisfy all criteria in the selection process, as shown in Table 4.

Table 4 Inclusion and exclusion criteria for study selection

Code	Description
IC1	Studies published between January 1995 and January 2026
IC2	Studies written in the English language
IC3	Peer-reviewed journal articles, conference proceedings, or book chapters
IC4	Studies relevant to the research questions and domain keywords
IC5	Studies meeting the predefined quality assessment criteria
EC1	Studies published before January 1995 or after January 2026 were excluded
EC2	Studies written in languages other than English
EC3	Non-peer-reviewed literature (e.g., reports, magazines, opinion pieces, editorials)
EC4	Duplicate studies or repeated publications of the same work
EC5	Studies failing to meet the quality assessment standards

4.4 Quality assessment criteria

In line with the SLR methodology, a quality assessment of the selected studies was conducted to ensure the review's validity and reliability. The main aim of the quality assessment was to include only high-quality studies that directly address the research questions and minimise potential biases in study selection. The SLR process enables systematic quality checks, supporting an objective and transparent decision-making process (Asadi et al. 2019).

We established an objective metric to assess the paper's quality, ensuring that its findings are reliable and accurate. We prepared a list of four quality assessment questions with relative scores that could provide a reliable indication of the quality of the reviewed papers.

The quality assessment included only four questions, as this SLR covers the specialised research area, such as bitemporal databases. A more elaborate rubric risks introducing inconsistency due to the diversity of studies spanning theoretical models, system implementations, and subjectivity across different application-based research. Moreover, this approach aligns with established SLR guidelines (Farina et al. 2022), which recommends avoiding overly complex scoring schemes to maintain consistency and reliability in evaluations.

QA1: Does the study effectively address the research questions related to bitemporal databases?
QA2: Does the study clearly describe its research objectives and methodology?
QA3: Are the objectives achieved, and are the results clearly presented?
QA4: Does the study make a valuable contribution to the field of bitemporal database research?

A quality assessment of the 248 studies retrieved from all four databases using the search strategy described in this review indicates that the quality of the evidence is strong. Each study was evaluated using four quality assessment questions. The total possible score was 4 points. Fully satisfied criteria question was 1 point, partially satisfied criteria received 0.5 points, and unmet criteria received 0 points. Studies with a total score greater than 2 were approved, whereas those with a score of 2 or lower were rejected. This assessment ensured the rigour and reliability of the studies included in the review and guaranteed that only high-quality research was considered for further analysis and discussion. All selected primary studies were assessed using these criteria on an ordinal scale (e.g., Yes = 1, Partial = 0.5, No = 0). Studies with quality scores of 50% or higher have been considered quality studies for this SLR. Appendix 1 provides the quality score for each primary study.

4.5 Conducting the review

Primary study selection The PRISMA framework was used to identify, screen, and include the most relevant literature for this SLR (Singh and Chaturvedi 2021). As shown in Fig. 2, the study selection process has followed the predefined protocol to uphold the quality of this review. The study selection and quality assessment were conducted by the primary author and verified by co-authors. In cases of disagreement, decisions were resolved through discussion to reach consensus, in line with established SLR guidelines (Hilmi et al. 2023).

The study selection process was divided into four steps: identification, screening, eligibility, and inclusion. Our initial query returned 248 studies from IEEE, Scopus, ProQuest, and Web of Science. At the screening stage, 53 duplicate records were removed. The remaining 195 studies were screened based on predefined inclusion and exclusion criteria, resulting in 148 studies. Subsequently, during the eligibility stage, titles, abstracts, and keywords were reviewed, resulting in the exclusion of 53 studies, leaving 85. Following full article assessment, 18 studies were removed because they did not focus on bitemporal databases, leaving 67 studies. In the final inclusion stage, all 67 studies underwent a structured quality assessment. Therefore, 13 studies did not meet the minimum quality threshold of 50% and were excluded. Finally, 54 studies were included in this SLR for data extraction and synthesis.

4.6 Data extraction

Data extraction is a crucial part of an SLR. It involved selecting and recording relevant and important information from primary studies. It provides the basis for evaluating, analysing, summarising, and interpreting the evidence (Taylor et al. 2021). Table 5 summarises the data items extracted from each study, along with their descriptions and mappings to the corresponding research questions. Using this framework,

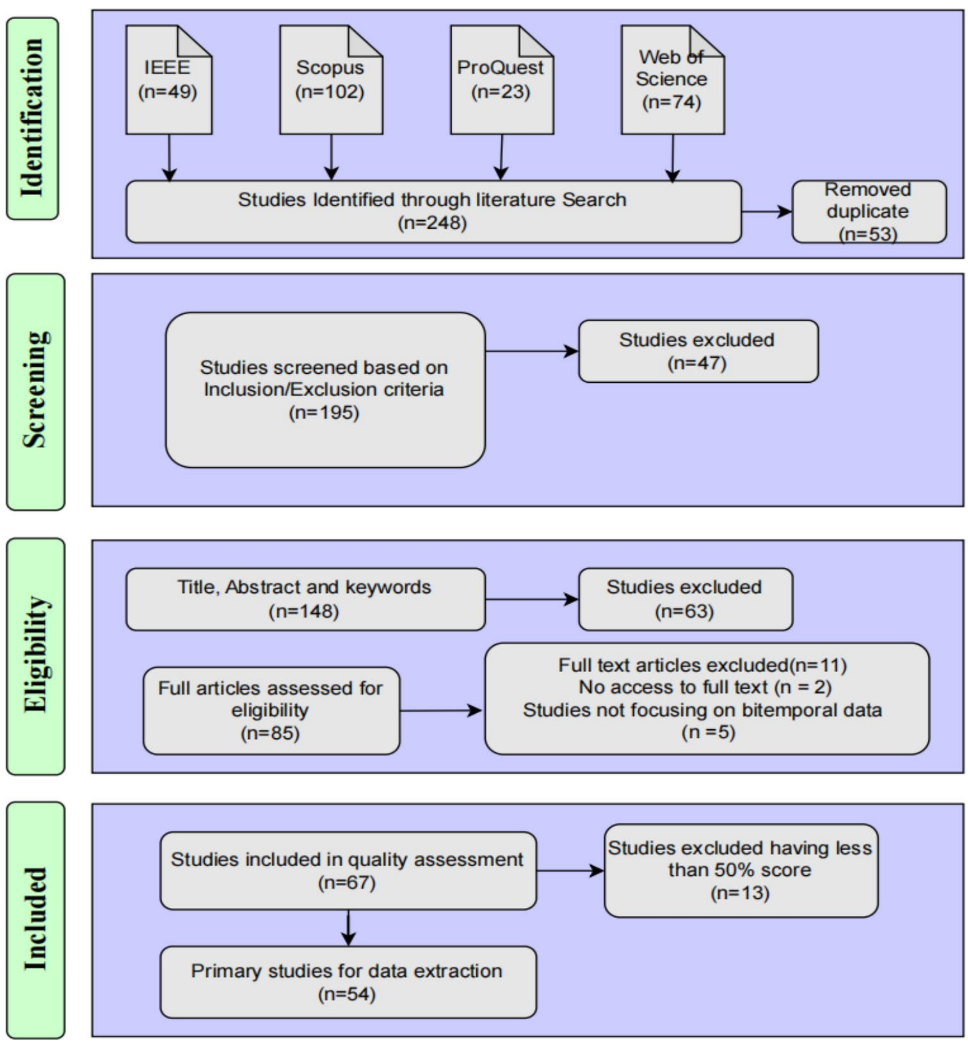


Fig. 2 PRISMA framework

relevant bibliographic, methodological, and thematic information was extracted from the selected studies. The extracted data served as the basis for addressing the research questions and for conducting the subsequent analysis and synthesis.

4.7 Data synthesis

Data synthesis is used to collect and summarise the data extracted from primary studies. Moreover, the main goal is to understand, analyse, and extract insights into how bitemporal databases have been explored in the existing literature. Our data synthesis is specifically divided into two main phases. In the first phase, we analyse the extracted data, most of which are included in Table 5 to determine trends and to collect information on our research questions and themes related to bitemporal databases. In the second phase, we classify the literature by research question. The most important task is to classify articles by model, use case, and key challenge, using relevant analysis. This classification helps to clarify how different studies contribute to the research landscape and future directions.

Table 5 Data extraction table

Data items	Extracted data	Description	Type
1	Title	Reflect the research direction	Whole research
2	Year of Publication	Indicating the trend of the research	Whole research
3	Journal/Conference	Indicating the type of study	Whole research
4	Authors	Research the authors' relevant study	Whole research
5	Key Challenges	The limitations of approaches and challenges of bitemporal databases	Whole research
6	Publication patterns	Trends such as yearly growth	RQ1
7	Research themes	Main research directions (modelling, querying, applications)	RQ1
8	Bitemporal model type	Type of model used: logical, interval, point-based	RQ2
9	Temporal semantics	Rules for handling time-sensitive data corrections	RQ2
10	Use cases	Domains where bitemporal databases are applied	RQ3
11	Implementation details	DBMS or system architecture used (relational, NoSQL)	RQ3
12	Future research directions	Proposed improvements and open research opportunities	RQ4

The methodology of this study is designed as a hybrid systematic literature review approach. The first layer consists of bibliometric and scientometric analyses, which support the overall review rather than serving as independent methods (Hilmi et al. 2023). It employs VOSviewer and BibExcel as tools for bibliometric and scientometric mapping (Doulani 2021). The second layer's primary contribution lies in the qualitative synthesis of the selected literature, including a comparative evaluation of bitemporal models, an assessment of DBMS infrastructure support, analysis of domain-specific applications, and the identification of research gaps. The second layer applies the PRISMA framework and the GQM approach as methodological frameworks to guide the qualitative synthesis and systematic data extraction from the 54 included studies (Farina et al. 2022).

Thus, integrating these two layers in a hybrid approach enables a more comprehensive understanding of the field by linking macro-level research patterns to micro-level technical insights. Such combined approaches are increasingly recognised in systematic literature review practices, where bibliometric techniques are employed to strengthen, rather than replace, structured qualitative synthesis (Ding and Meng 2014b).

5 Results and discussion

In this section, we present the quantitative and qualitative overview of the review, and we align our results by addressing the four research questions as shown in Table 2.

5.1 Scientometric and bibliometric analysis

This subsection presents the scientometric and bibliometric results that address RQ1. It covers topics such as publication patterns, influential contributors, keywords and themes in the field of bitemporal databases.

5.1.1 Publication trends over time

Publication trend analysis is a common element of bibliometric reviews, as it shows how research output in a specific field has developed over time and highlights periods of notable growth or decline (Donthu et al. 2021).

Figure 3 presents the distribution of all studies from January 1995 to January 2026. In the early phase (January 1995 – December 1999), the number of publications remained relatively low; however, an initial upward trend was observed at the end of December 2000. Interestingly, the number of publications dropped dramatically in 2009. A second significant upward trend in publications was observed in 2011. It is worth noting that the initial upward

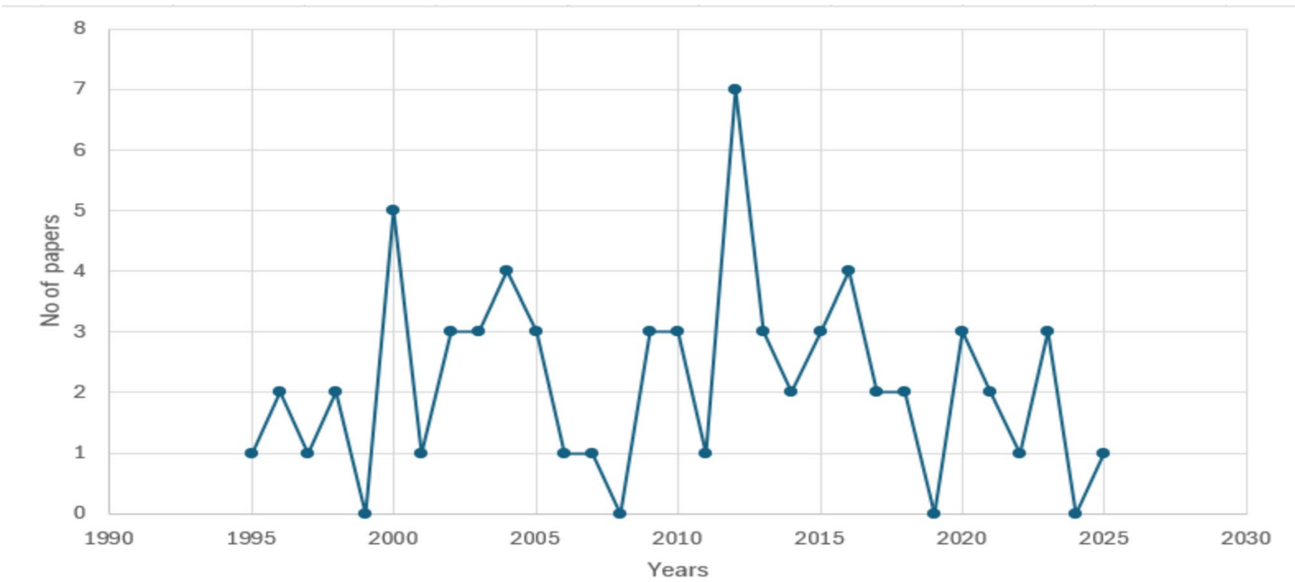


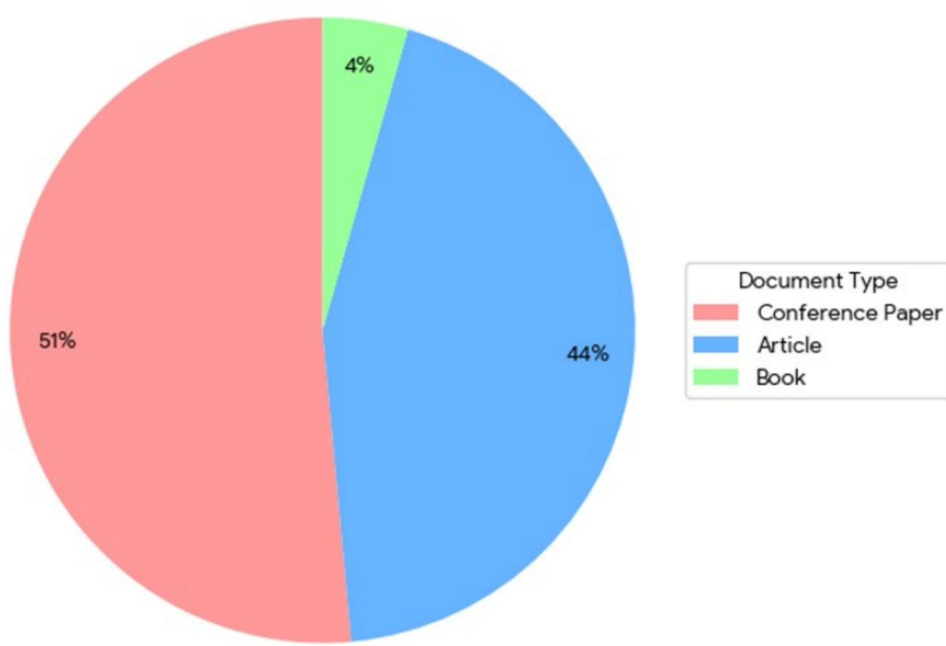
Fig. 3 Distribution of publication years

trend, largely driven by the collaboration between Jensen, Christian, and Snodgrass over a five-year period, focused on the semantic understanding of temporal databases. Their effort contributed significantly to subsequent growth in the field of research (Debrouvier et al. 2021). However, prior to 2011, the bitemporal research field was mostly concentrated on academic concepts that were difficult to implement in practice; however, the built-in support for bitemporal data in the SQL:2011 standard, which was introduced, bridged theory and practice and led to a rapid expansion of related publications, as shown in Fig. 3.

5.1.2 Type of publications analysis

As previously stated, this review comprises 54 academic papers (including conference proceedings, journal articles, and book chapters). Figure 4 shows that most papers in the sample were published in academic journals (51%) and conference proceedings (44%), with only a few (4%) appearing in book chapters. This shows that the topic is primarily disseminated through peer-reviewed journals and conference venues, indicating ongoing development of new findings. The scarcity of book chapters may indicate that the

Fig. 4 Distribution of the types of literature



field remains specialised and has not yet reached broader audiences.

The distribution of published studies indicates that most research articles are published in IEEE Access (19 articles), followed by Springer Nature (16 articles) and Elsevier (8 articles). Furthermore, ACM (Association for Computing Machinery)also ranks among the top publishing sources, contributing 7 papers, as shown in Fig. 5. The number of research papers in the bitemporal field has gradually increased since 2011, while the number of book chapters has remained stable.

5.1.3 Cross collaboration analysis

Figure 6 presents a Three-Field Plot showing the relationships among authors, their affiliated countries, and the most frequently used keyword. The figure was mapped using Bibexcel, a bibliometric tool developed specifically to manage bibliometric data and build maps (Doulani 2021). Regarding geographic distribution, many studies originate from the USA and China, owing to their leading research universities, substantial funding for database infrastructure, and high publication output. Denmark makes a notable contribution, reflecting the early and ongoing influence of foundational research conducted at Aalborg University. This is due to the historical impact of Professor Jensen and Snodgrass's bitemporal foundational collaboration at that university. Furthermore, countries such as Turkey and Italy demonstrate consistent contributions, which can be linked to active research groups focusing on areas such as temporal data modelling, semantic web technologies, and XML-based temporal systems. Overall, the distribution highlights both the concentration of research in established scientific hubs and the emergence of contributions from diverse regions.

This shows the level of collaboration between authors across countries. The distribution of keywords such as valid time, temporal model, access methods, data models, and bitemporal data reveals a sustained research emphasis on modelling both real-world time dimensions in bitemporal data management. Authors such as Tang, Kaufmann, Christian, and Snodgrass appear within this collaborative network, suggesting their influence in advancing the bitemporal databases domain. Furthermore, keywords such as data models, architecture, access methods, and semantics indicate that continuous work has been done to represent querying and enforce temporal constraints. Overall, the analysis shows that research on the bitemporal domain is globally distributed, indicating a combined effort by scholars across sectors and research fields to develop robust frameworks for bitemporal database management, thereby justifying the need for this SLR study.

6 Keyword co-occurrence analysis

Keywords are terms or phrases that represent the main content of a document and articulate the research area within specific domain boundaries. In this study, VOSviewer software performs a keyword co-occurrence analysis and generates networks from database data, illuminating key research themes (Klarin 2024). In total,1717 keywords have been obtained. The minimum number of keyword occurrences is 8. A total of 323 keywords met the criteria, of which 117 were used for analysis. Figure 7 shows the keywords co-occurrence

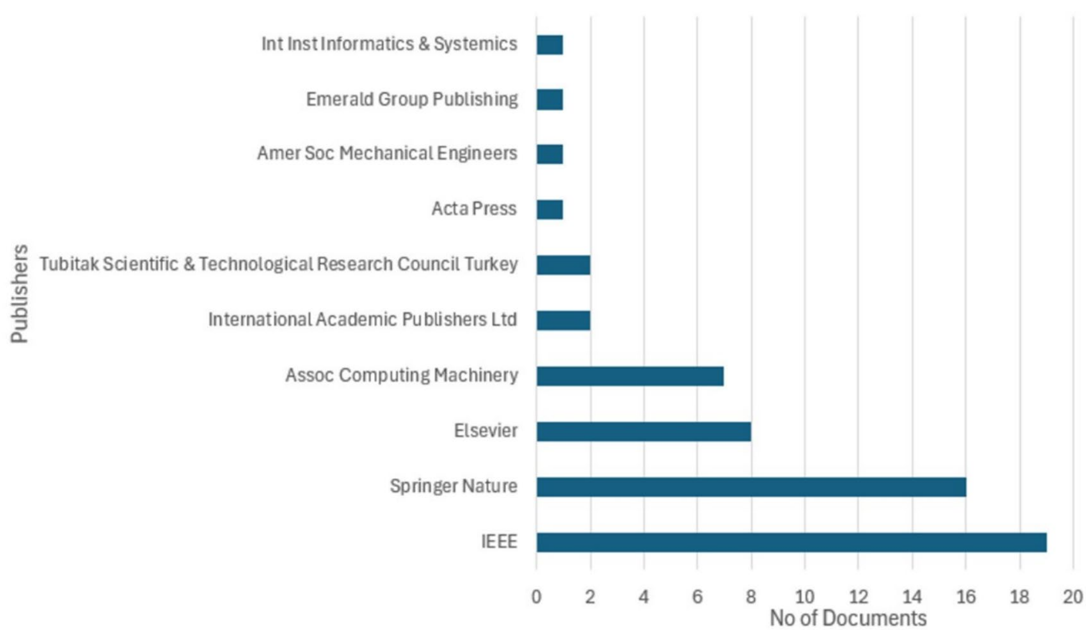


Fig. 5 List of top 10 publishers

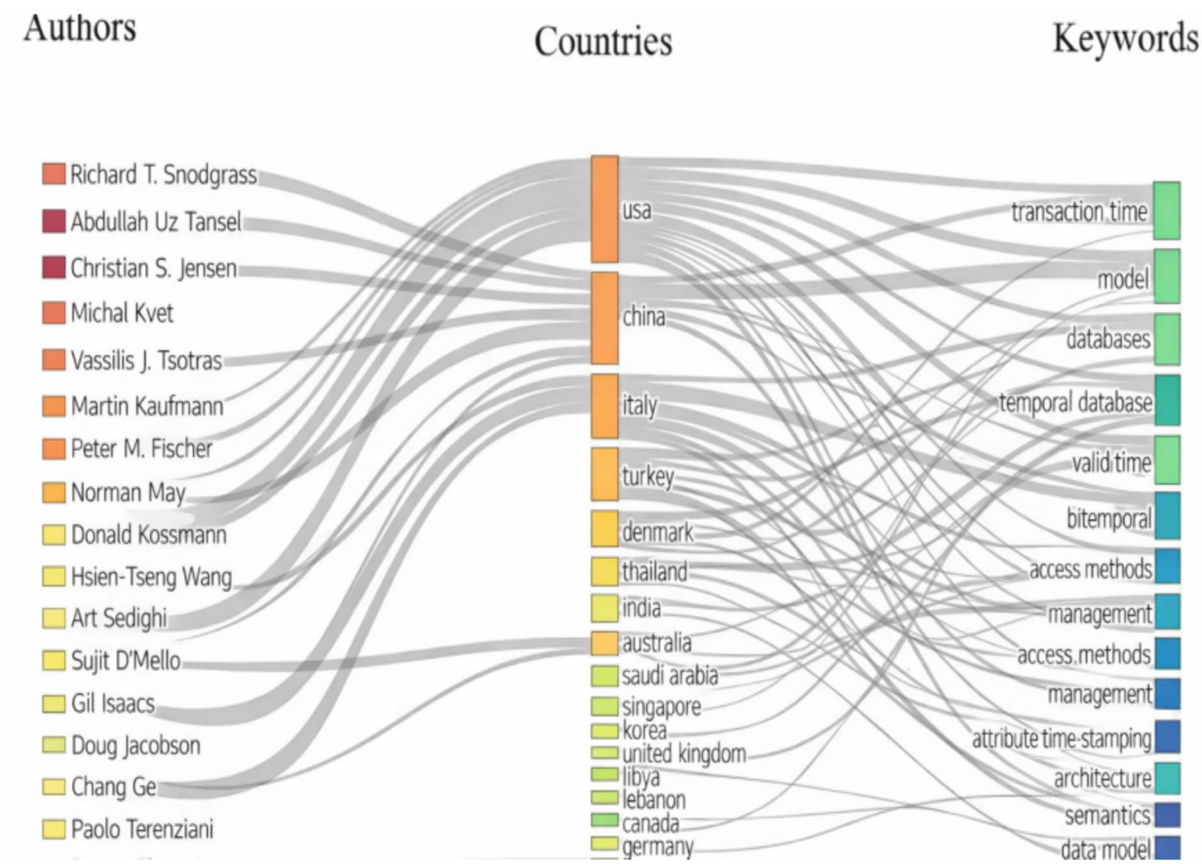


Fig. 6 Three-field plot illustrating bibliometric mapping of authors, countries, and research keywords

network with 23 nodes, and a total link strength of 102. Table 6 summarises the four main cluster keywords. Each reflects distinct thematic areas within the field.

Table 6 was constructed using VOSviewer (Klarin 2024) and provides four quantitative metrics for each cluster: occurrence counts, number of links, mean publication year, and total link strength. These metrics play a vital role in analysis; occurrence counts show how often a research theme appears across the 54 included studies, link counts indicate the strength of a cluster's connection to other thematic areas, mean publication year reveals whether a theme is historically established or recently emerging, and total link strength measures the overall intensity of co-occurrence relationships within the cluster (Ding and Meng 2014a). Together, these four metrics provide the quantitative foundation for the thematic interpretations of each cluster, ensuring that the cluster descriptions are evidence-based rather than speculative.

The green cluster forms the theoretical and historical core of all research networks. It shows the highest level of activity, with 52 occurrences and the strongest link strength of 42. They are its foundational concepts. Its most significant and most central node, bitemporal data, anchors this cluster along with its parent concept, temporal databases. This group primarily deals with the principles of data management over time. Keywords include data model, indexing, query processing, relational databases and timestamping (Kleiner and Lipeck 2002). All these terms describe the fundamental problems that researchers have attempted to address: How to model, store, and query data with dual dimensions: valid time and transaction time (Trudel and Goodwin 2000). The term now points to specific, complex problems in temporal databases, such as queries relative to the current moment. The theory and architectures that form this cluster are the foundation of the field. The publication year of 2023 confirms that this is a more recent research direction.

The Red Cluster is the most modern, application-driven research front in bitemporal. It consists of 35 occurrences and a link strength of 28. Its major theme is changing detection, mainly applied to remote sensing images (Chen et al. 2024). This cluster applies advanced computational techniques. Keywords such as deep learning, feature extraction, convolutional neural networks (CNNs), and computer vision dominate this group. This means the work primarily focuses on sophisticated algorithms for detecting temporal changes in satellite or aerial imagery. Terms such as unsupervised, semi-supervised, and object detection further reflect the cutting-edge nature of

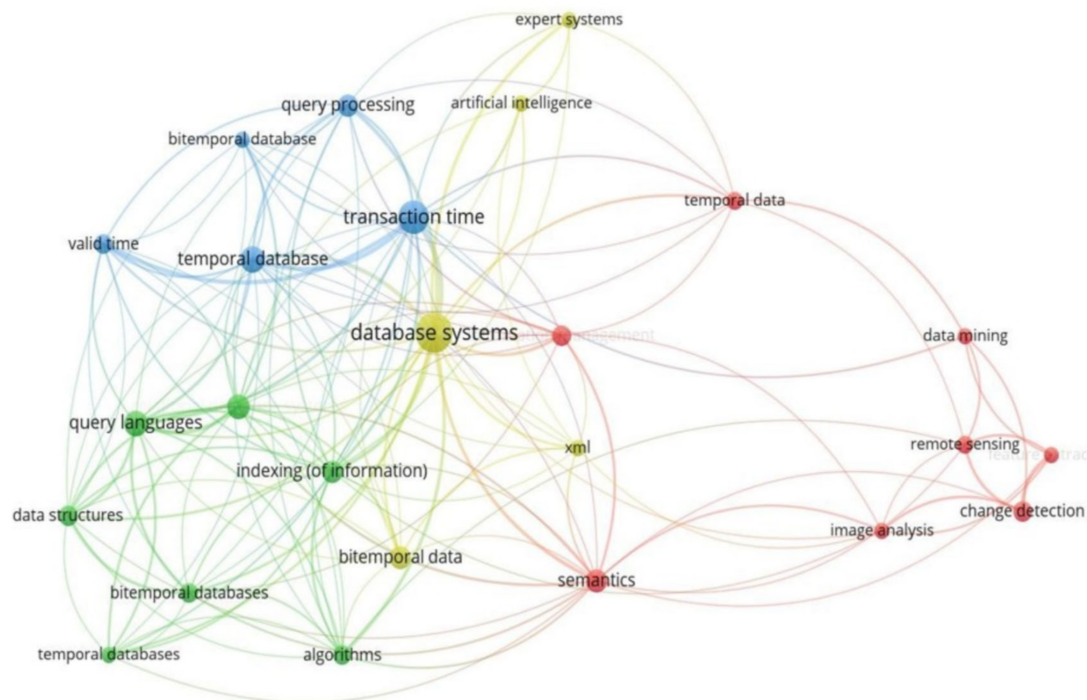


Fig. 7 Keyword co-occurrence network

Table 6 Cluster information with keywords and metrics

Cluster	Key keywords	Occurrences	Links	Mean year	Total link strength
Red	deep learning, CNN, computer vision, feature extraction, object detection	35	12	2023	20
Green	bitemporal data, temporal databases, data model, indexing, query processing, valid time, transaction time	52	28	2018	42
Blue	data warehousing, data modelling, XML, semantics, knowledge graphs, fuzzy logic	29	15	2015	24
Yellow	benchmarking, performance, evaluation, workloads, query algorithms	23	10	2019	18

this research, which aims to eliminate manual data labelling (Atay and Alp 2016). This cluster represents a significant step away from traditional database theory toward a concrete, high-impact application domain that utilises artificial intelligence to analyse large-scale geospatial data. It is important to note that these keywords emerged from the broader bibliometric keyword pool across the four databases and reflect co-occurring research themes in the literature landscape. Studies that focused purely on image processing or computer vision without a bitemporal database component were identified and excluded during the eligibility screening stage of the PRISMA process, ensuring that the 54 included studies maintain a clear focus on bitemporal data management principles.

The blue cluster bridges foundational theory with structured applications, typically in business and information systems and with 29 occurrences and a total link strength of 24. Keywords such as data warehousing, which maintain a complete time-variant historical record for business intelligence. This cluster also focuses on data modelling and structure, with keywords such as XML, semantics, knowledge graphs, and entity-relationship models (Wang and Zaniolo 2004). This suggests a focus on producing rich, semantically meaningful historical datasets beyond relational tables. The inclusion of fuzzy logic and clustering suggests research into uncertainty handling and the organisation of temporal information (Fredrick and Radhamani 2015). Essentially, this cluster concerns the practical implementation and conceptual organisation of bitemporal data in complex information systems, allowing historical data to be stored and analysed.

The Yellow Cluster, a smaller but critical group, has 23 occurrences and a link strength of 18. This cluster study demonstrates how bitemporal systems perform under real-world workloads, including storage structures, indexing techniques, and query algorithms (Kaufmann et al. 2015). That sort of evaluation enables the field to mature by enabling objective comparisons of approaches and by identifying performance bottlenecks and opportunities for improvement in bitemporal database technologies.

Thus, the cluster results suggest that bitemporal database research extends beyond theoretical advances in temporal models to address practical querying and system-implementation issues. Clustering these keywords suggests a robust interaction between foundational theory and practical system research in the field.

Overall, the bibliometric findings in Section 4.1 directly address RQ1, and the identified cluster themes, including foundational modelling, application-driven research, data warehousing, and system performance, provide the thematic foundation for the qualitative synthesis presented in Sects. 4.3 through 4.7, linking the bibliometric evidence to RQ2, RQ3, and RQ4 (Klarin 2024).

6.1 Thematic map analysis

The previous stage analysis of keyword co-occurrence identifies how research topics cluster based on their frequency and relationships; it does not reveal the developmental status or strategic importance of those themes within the field. To address this gap, a thematic map was generated.

A thematic map is a science-mapping visualisation that displays clusters of related keywords in a two-dimensional space defined by centrality and density. Its synthesis categorises studies and summarises findings by identifying major or recurrent themes within studies generated using bibliometric tools, providing valuable insights into the conceptual structure of a field and the relationships among its thematic areas (Ding and Meng 2014a).

Themes are organised into four quadrants: Basic themes, Niche themes, and Emerging or Declining themes.

Basic themes with high centrality but low density exhibit broader research-bubble networks. During this phase, Big Data is a prevalent concept (Zhang et al. 2024).The study suggests that this quadrant is mainly linked to large-scale data management and temporal challenges, where studies consistently recognise the importance of handling high-volume, high-velocity temporal data, such as benchmarking, DBMS support, and system-level implementation, with examples like TPC-based benchmarks, PostgreSQL temporal extensions, and cloud systems such as CosmosDB. However, the study remains largely theoretical, despite ongoing DBMS support for bitemporal functionality since the introduction of standard SQL:2011 (Kaufmann et al. 2014). It often necessitates custom extensions or workarounds. This results in a lack of widely accepted evaluation frameworks for temporal performance, scalability, and correctness, suggesting that it has not yet achieved the same scope as other models (Petković 2017).

On the other hand, the motor themes represent the most established and actively developed area of the field, as they show both high density and high centrality, indicating strong internal development and close interaction with other research clusters. In this study, the top-right quadrant of the thematic map is dominated mainly by core bitemporal modelling and query language research, including concepts such as bitemporal structured query language (BtSQL) and XML-based temporal models (Wang and Zaniolo 2004). These themes essentially form the technical backbone of bitemporal database research, as well as the well-defined semantics and querying mechanisms that underpin contemporary developments. However, as these themes are methodologically mature and internally coherent, they largely focus on temporal modelling and query representation, with limited progression towards large-scale integration. This indicates that advances in formalism are unlikely to produce equivalent progress in implementation or real-world use cases.

The niche themes can be seen in the top left quadrant of the Thematic map graph. These are very dense, indicating high internal strength, and low in centrality, indicating little interaction with other clusters. In this study, they often aim to solve domain-specific problems, such as belief-based and graph-based bitemporal databases (Rost et al. 2021), However, their low centrality suggests these contributions remain within specialised research streams rather than being discussed in the main body of bitemporal databases. As a result, their overall impact on the field is still quite limited (Rost et al. 2021).

The emerging themes can be seen in the bottom left quadrant of the Thematic map graph. These are low in density and centrality, indicating that they neither have high internal strength nor interact significantly with other clusters. In this study, it includes areas such as time series forecasting, AI integration, and use cases (Zhang et al. 2013). These themes are quite important, as they seem to point towards where the field is heading, especially with growing data-driven decision-making in intelligent systems. However, since they are very low in both centrality and density, they are not yet well developed and are not strongly connected to the broader body of the bitemporal databases (Wondoh et al. 2015). They may also move to the niche theme or motor theme quadrants in the future, depending on how research around these clusters progresses. Thus, the thematic map shows the core key themes and their growing attention in bitemporal databases (Fig. 8).

Overall, the results from the publication trend analysis, three-field plots, keyword co-occurrence analysis, and thematic maps consistently reveal that bitemporal database research has steadily increased in volume but unevenly in depth. Theoretical foundations, particularly regarding valid time, transaction time, and temporal modelling, remain well established and are

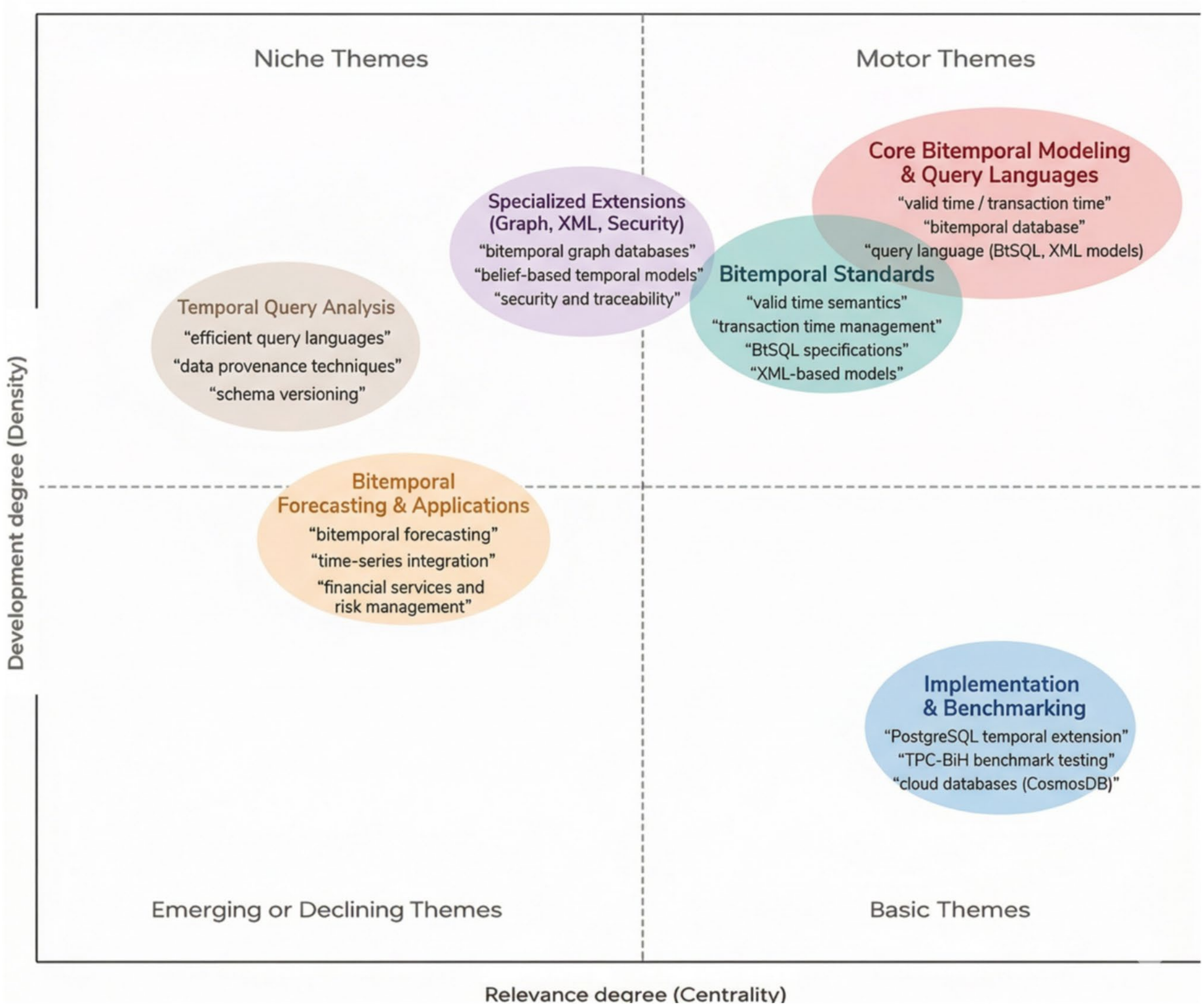


Fig. 8 Bitemporal thematic map

continually reinforced. This may be due to the SQL:2011 standard, as it led to a sudden rise in the number of studies, providing researchers and practitioners with a shared implementation reference and expanding the field into a wider range of applied studies (Kulkarni and Michels 2012). Meanwhile, the three-field plot indicates that this development has been concentrated among a limited number of influential countries, authors, and core keywords, especially in areas such as valid time, temporal modelling, and access methods (Doulani 2021).

The keyword co-occurrence analysis shows that foundational modelling and indexing concepts remain the most strongly connected areas of the field, while application-oriented and performance-focused themes are more dispersed and less central. The thematic map shows that areas such as Artificial Intelligence are emerging but not yet fully implemented. This suggests that, although the scope of bitemporal research is expanding, its methodological development has been comparatively limited. Thus, these observations indicate that bitemporal research has moved beyond its initial focus on foundational semantics. However, its progress towards scalable infrastructures, broader cross-domain adoption, and mature real-world applications is accelerating, even though it remains incomplete. This supports the more detailed comparative analysis in Sects. 4.3 to 4.7 and underscores that the field is characterised not only by growth but also by ongoing tension between theoretical development and practical implementation.

6.2 Modelling and implementation approaches

This subsection addresses Research Question RQ2 by examining how different bitemporal database models and implementations represent bitemporal data in query processing and in the interpretation of valid-time and transaction-time semantics.

Bitemporal data modelling has several dominant approaches (Won and Elmasri 1996). Note that different modelling techniques have been developed on bitemporal principles adapted to various application needs and system environments (Atay and Tansel 2009). We analysed and systematically classified the main existing models for bitemporal data management, as presented in Appendix 2. This comparison of different models helps to understand how each model addresses the problems, which query language it utilises, its application sectors, its limitations, and how it represents bitemporal functions within database systems.

6.2.1 Object-oriented bitemporal models

Object-oriented databases are used to handle bitemporal data by encapsulating time within objects, for example, in the bitemporal object, State, and Event Modelling Approach (BOSEMA) by Njovu and Ibrahim, which defines design patterns for reusable and modular bitemporal Object Modelling in DSSs (Decision Support Systems) that require historical data along with current data for retrospective and predictive analysis. This avoids the complexity of capturing the full temporal semantics of data in database models. It defines a framework for enriching data modelling with a core model of typical bitemporal object types for each bitemporal database model. This model has the advantage of capturing the basic temporal semantics of an ordinary bitemporal database object that designers can extend to capture their data needs in a modelled bitemporal app domain (Njovu and Ibrahim 2003). However, BOSEMA is a modular, reusable framework for bitemporal object modelling in DSSs, focusing on design patterns and core object types of definition rather than query processing or indexing strategies.

El Hayat et al. (2020) address that standard UML lacks mechanisms to represent time-varying data and temporal constraints. The authors proposed a UML/OCL model that supports valid-time and transaction-time at the attribute level by improving clarity in bitemporal schema design. This approach contributes to conceptual modelling and framework for bitemporal design in object-relational databases; however, it does not cover query processing or indexing solutions for bitemporal databases, as it focuses more on access efficiency rather than temporality modelling.

In the POINT approach,"now" is represented as points on time axes and logical query transformations are applied to optimise access to bitemporal data. This method allows the use of standard spatial indexes, such as B+-tree/R*-tree, without modifying a database engine. Its experimental results outstrip traditional maximum-timestamp methods by more than 10 × in disk and CPU efficiency. However, it is essential to emphasise that the advantages of the point approach are mainly independent of the specific indexing methodology used. To substantiate the claim, they extend the experimental work by comparing the POINT and MAX approaches in case conventional B+-tree indexing is adopted (Stantic et al. 2009).

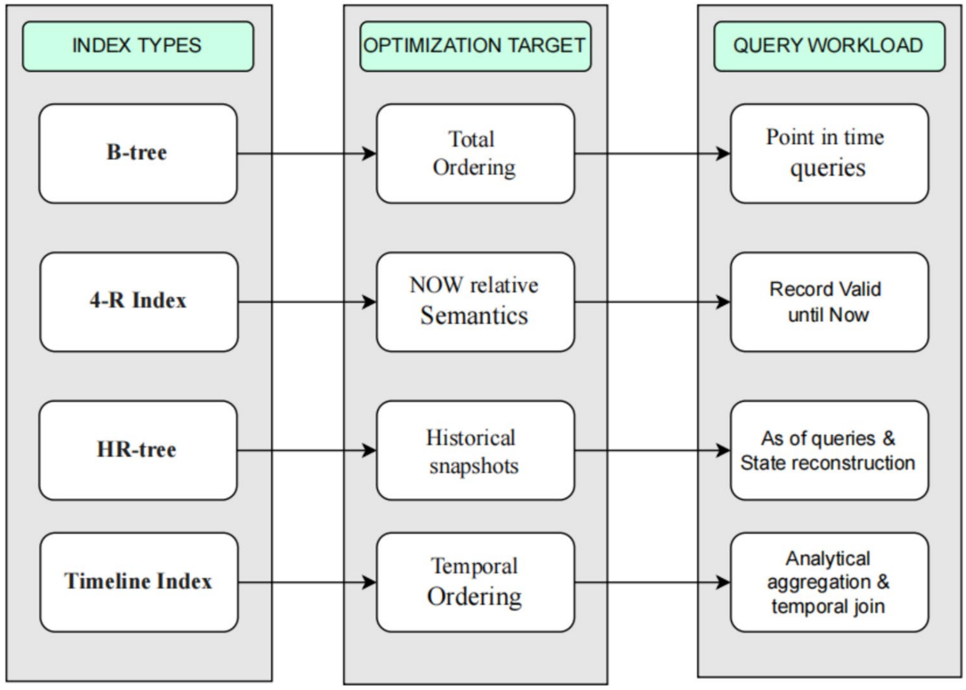
6.2.2 Indexing and optimization models

The bitemporal Timeline Index model exploits modern hardware characteristics, such as large main memory and fast sequential scans. Instead of modelling the dual dimensions, valid-time and transaction-time, as 2D spatial structures, they model them independently, with a dedicated one-dimensional index for each temporal domain. The combined application index appeared only at specified checkpoints, with minimal computational overhead. This lightweight strategy enables operations such as selection, joins, and temporal aggregation to outperform traditional spatial and temporal Index structures (Nascimento et al. 1996). The approach also includes index maintenance algorithms that balance space consumption, update costs and query performance while supporting uniform implementations for temporal operations across time dimensions (Kaufmann et al. 2015).

Agesen et al. (2001) propose a system that, for now, splits the operator for relative bitemporal databases to address issues introduced by the dynamic NOW timestamp across dual dimensions, such as valid and transactions. It was done by breaking down bitemporal areas into smaller, overlapping temporal segments, enabling complex bitemporal operations to be expressed using standard relational algebra. The method increases scalability by preserving timestamp identity and point-based semantics. Thus, the authors split the operator by providing formal definitions of bitemporal difference and aggregation, supported by SQL implementations and performance evaluations.

Figure 9 illustrates the indexing mapping techniques used in bitemporal databases, showing how each index is designed to support specific query types and optimisation goals. Traditional B-tree indexes are built for one-dimensional ordering and are suitable for simple snapshot queries that retrieve data at a single point in time (Eid 2002). However, they are not well-suited for temporal interval queries, such as overlapping valid-time or transaction-time periods, because they do not natively handle multi-dimensional temporal conditions. In contrast, the 4-R index highlights now-relative data that rapidly processes queries over valid time intervals by introducing a bitemporal function into fixed areas (Stantic et al. 2004). The advantage of this approach is that the database system does not require internal modifications to process time-based queries. The HR-tree accesses historical database states by sharing physical storage and maintaining multiple logical versions of the data over time, which supports frequent As-of queries and state reconstruction (Saltenis and Jensen 2002). Overall, the indexing layout

Fig. 9 Mapping of bitemporal index types to optimization targets and query workloads



explains how different indexing approaches address temporal database patterns, demonstrating that no single index structure is optimal for all bitemporal query workloads.

6.2.3 Schema versioning and temporal graph models

We recognise that schema versioning and the temporal graph model are becoming two key themes in bitemporal databases, as schema versioning first appeared in the late 90 s, with foundational work indicating that systems need both data versioning and schema versioning to preserve historical data (Snodgrass 1990).

By examining the limitations of existing single-table and multiple-table versioning strategies, Wei et al. (Wei and Elmasri 2000) proposed a schema versioning approach for bitemporal databases. The proposed method addresses issues associated with partial multiple table versioning by reducing storage overhead and data duplication. While preserving bitemporal semantics, the approach significantly improves query processing performance and schema conversion efficiency. The multi-temporal XML databases method is used to maintain previous schema versions and XML documents. This method is based on SML schema languages (XSD) and views schema modifications as a versioning procedure rather than straightforward evolution, and it enables querying across multiple schema versions using temporal extensions to XQuery (Brahmia and Bouaziz 2008).

Alongside schema versioning, our review shows a significant rise in bitemporal graph models, which are commonly used in modern applications such as IoT, knowledge graphs, digital twins, and cybersecurity. Relational bitemporal models struggle to efficiently represent evolving graph structures, but our review identified a recent work by Rost et al. (2021). Challenging this limitation, the authors propose the Bitemporal Property Graph Model (TPGM+), which supports bitemporal functionality within property graphs by introducing temporal extensions to PGQL (Property Graph Query Language) that enable expressive historical graph queries. Time-based models proposed by Saroha et al. (Saroha and Gosain 2016) capture both the valid and transaction-time dimensions by allowing multiple schema versions to coexist within the same valid period. This study focuses mainly on schema and dimension evolution, rather than query processing or performance optimisation. Wang et al. (Wang and Tansel 2019) work with semantic web data by extending RDF with dual time dimensions; their model allows multiple temporal states with resources and relationships to coexist with the same RDF structure. Their papers propose a formal framework for theoretical modelling and semantic consistency for future investigation.

6.3 Bitemporal support infrastructure

This section discusses the DBMS support for bitemporal infrastructure used or proposed in the SLR studies. Table 7 Timeline of DBMS CategoriesTable 7 shows different DBMS Categories along with their respective years of activity.

Within relational DBMSs, the use of classical relational engines can be extended due to SQL:2011 temporal features. Several DBMS vendors, such as IBM DB2, Oracle, SQL

Table 7 Timeline of DBMS categories

Category	Technologies	Years
Relational	DB2, Oracle, SQL Server, MySQL, Teradata	1998–2014
Non-Relational	MongoDB, Cassandra, NoSQL	2016–2018
XML/ Semi-Structured	XML DBs, Object DBs	2003–2008
Graph	Neo4j (temporal extensions)	2021–2025
Hybrid/Polyglot	SQL + NoSQL, Hadoop + RDBMS	2021–2022

Server, and MySQL, provide native or semi native bitemporal functionality (Kaufmann et al. 2014). On the other hand, open-source giants such as PostgreSQL rely on extensions or complex user-space logic to achieve similar functionality. However, managing dual temporal dimensions in relational systems introduces performance overhead, particularly for retroactive updates that require cascading interval adjustments across historical records. Furthermore, despite the SQL:2011 standard, the degree of native bitemporal support varies considerably across vendors, such as IBM DB2 and Oracle, which offer relatively complete support, while MySQL and PostgreSQL rely on partial implementations or user-defined extensions (Petkovi´c, D. 2017).

In the case of non-relational DBMSs, Eshtay et al. (2019) address the lack of native bitemporal support in NoSQL databases and propose schema-level modelling strategies to embed dual time dimensions in key-value and column-oriented NoSQL stores via explicit timestamp attributes and composite keys. This approach is illustrated using Redis and Cassandra in a bitemporal manner without requiring changes to the underlying NoSQL engines. A key practical limitation of this approach is storage duplication, as embedding explicit timestamp attributes significantly increases data volume compared to systems with native temporal operators, creating additional overhead for high-frequency data environments such as IoT and financial transaction systems (Al-Romema and Nashwan 2010).

Wang and Zaniolo (2004) model demonstrates an XML structure that utilises hierarchical data models to support bitemporal data, grouping timestamped elements within a document. This design allows bitemporal queries to be expressed using standard XQuery. Using this structure, dual time semantics can be reconstructed. However, most commercial XML systems rely on user-defined data models. Existing research indicated that an XML-oriented system can represent bitemporal histories more naturally than traditional SQL tables by utilising custom query patterns to manage temporal data effectively.

Rost et al. (2022) graph databases support bitemporal data with limited features; for example, in common property graph platforms such as Neo4j, temporal information is typically stored as timestamp attributes on nodes or edges. While this allows basic time-based filtering, the lack of dedicated temporal indexes and operators results in poor query performance, further underscoring that early temporal graph approaches mainly relied on complex extensions rather than built-in temporal semantics. To overcome these limitations, the authors suggest a distributed graph analytics framework that extends the property graph model with explicit support for valid-time and transaction-time, enabling comprehensive bitemporal graph queries.

In addition to relational, XML, and graph-based systems, few studies examine bitemporal support implemented at the DBMS architecture level using version management. Gancarski et al. (1999) propose an object-oriented approach based on the Database Version (DBV) model, in which valid time and transaction time are attached to complete database states rather than to individual tuples or objects. This allows multiple versions of objects to be maintained efficiently and enables correcting past data while keeping earlier records unchanged. The model also supports branching valid-time histories, which is difficult to achieve in conventional temporal designs. By separating logical database versions from physical storage, bitemporal functionality is handled within the DBMS itself rather than through query-layer extensions or additional indexing structures. This work shows that version-based mechanisms can provide native bitemporal support while improving historical consistency and flexibility.

Following this, Hou et al. (2304) identify two primary approaches to managing temporal graphs using a hybrid storage method. The first stores valid time intervals as attributes of nodes and edges, while the second represents graph evolution by combining multiple snapshots with delta updates. Their study shows that modelling timestamps as ordinary properties results in higher query costs as historical data accumulates, whereas snapshot delta methods increase storage overhead and demand costly reconstruction of past graph states.

Thus, our review identified a growing interest in hybrid infrastructure, in which both DBMS and non-DBMS support for bitemporal data are being researched. This hybrid approach reflects the need to accommodate transactional and analytical bitemporal workloads within a unified architecture.

6.4 Use cases of bitemporal databases

This subsection addresses Research Question RQ3, addressing the use cases of bitemporal databases.

Before examining individual domains, it is important to contextualise the scope of the available evidence. Of the 54 papers included in this review, only a limited number of studies address real-world application domains. This imbalance can be traced back to thematic map findings, where application-oriented themes occupy the emerging and niche quadrants rather than motor or basic themes, reflecting that application-oriented themes are still emerging and highlighting significant potential for further development and deeper integration within the broader research landscape.

Financial sector Bitemporal database models are increasingly required for financial applications to support historical data tracking and auditability. Research on the Polish merger market illustrates how bitemporal databases can address challenges posed by late-arriving information and produce results with greater precision earlier. The proposed model ensures the consistency and traceability of historical records of merger transactions despite delays in data updates across hardware and software systems, such as multicore processors, large-scale main memory, and new storage technologies. Bitemporal data management is increasingly indispensable for high-performance transaction processing, especially for applications that require tight auditing and compliance, such as financial risk analysis. Dual-temporal tracking provides full auditability and robust query performance across large, variable datasets with minimal lookup operations. These are critical capabilities for organisations managing high volumes of dynamic financial data under strict regulatory standards. Together, they highlight the growing need for bitemporal models in financial systems (Sedighi et al. 2020).

Supply chain sector The bitemporal database middleware indexing algorithm proposed by Yifei et al. (Du and Zhu 2024) suggests an indexing algorithm for e-commerce supply chain risk management. The proposed method addresses issues of delayed data reporting and inconsistent risk assessment. The system employs advanced indexing strategies for query execution, even in high data-velocity scenarios typical of e-commerce platforms. Additionally, incorporating digital marketing insights into the risk analysis pipeline helps the model understand supply chain dynamics and detect early disruptions driven by market behaviour. Experimental evaluation demonstrates that this method improves the precision and responsiveness of risk prediction models, making it a promising approach for intelligent and resilient supply chain decision-making.

Blockchain and cybersecurity sector Blockchain and cybersecurity are particularly relevant for bitemporal applications because both depend on tracking events over time. Our review identifies a recent paper that utilises BiTchain, an improved blockchain model that employs valid-time timestamps issued directly by data originators, such as sensors or vehicles. These timestamps record the precise moment an event occurs, reflecting real-world event timing. Furthermore, they extend the existing blockchain query language with a temporal extension of EQL to utilise the two temporal dimensions of transaction time and valid time (Brahmia et al. 2023).

Healthcare sector Temporal encoding follows bitemporal concepts, considering how medical data changes over application time and when it was recorded and updated over system time. Our review identified a paper in which the proposed model, known as BiteNet, produces a unified patient-journey representation using an end-to-end self-attention network to improve medical outcome prediction. This framework comprises a module that captures contextual information and temporal relationships between visits in a patient's healthcare journey, and an attention-pooling module that constructs a hierarchical structure of three-level EHR data. Furthermore, the model performs supervised prediction and unsupervised clustering using real-world ERH data. This shows that the BiteNet model can achieve superior performance and predictive power by integrating the bitemporal property (Peng et al. 2020).

6.5 Functional advantages of bitemporal databases

This section examines the key functional features of bitemporal databases that allow them to effectively address these needs across various application scenarios. Among the 54 papers reviewed, a few of the studies explicitly highlight the widespread adoption of bitemporal properties in areas such as data warehousing, auditing, and complex temporal analysis. Therefore, this subsection responds to Research Question RQ4 by summarising the functional advantages of bitemporal databases identified in the reviewed literature.

One of the most consistently reported strengths of bitemporal databases is their capacity to maintain temporal accuracy and representational flexibility across diverse data environments. Jensen and Snodgrass (1996) establish this through the Bitemporal Conceptual Data Model (BCDM), which preserves information content across different temporal states through snapshot equivalence, enabling retroactive corrections without loss of prior records. Stantic et al. (2009) further confirm this advantage in accuracy, demonstrating that point-based representation of the current timestamp outperforms traditional maximum-timestamp approaches by more than tenfold in both disk access and CPU efficiency, without requiring engine-level modifications. The bitemporal model allows multiple rows per object, distinguished by the object identifier and two-time intervals: A valid time interval and a transaction time interval. Such an approach captures the object's actual state over time, along with any changes or updates to that state (Anselma et al. 2010). For example, in bitemporal data analysis, Fuzzy Crisp Relative Spherical CURE Hierarchy Clustering (FCRSCHC) is a method for clustering temporal data that changes over time with minimal computational resources. It does so using minimal computational resources while demonstrating the effectiveness of bitemporal data analysis and data adaptability (Wang et al. 2013). Chau and Chittayasothorn (2021) similarly demonstrate this flexibility in SQL-based bitemporal design, where attribute-level timestamping allows the model to accommodate domain-specific temporal requirements without restructuring the underlying schema.

Beyond accuracy, the literature also identifies a key advantage, such as the ability of bitemporal databases to preserve and query both historical and projected data states. Johnston (2014) and Johnston and Weis (2010) demonstrate that dual-timeline management enables organisations to maintain complete non-destructive records across both operational and analytical workloads, offering a more robust alternative to slowly changing dimensions in data warehouse design. Bohlen et al. (2000) advance the research with temporal statement modifiers, which allow queries to retrieve data accurately at any point in time, thereby enriching the expressiveness of historical queries using SQL capabilities. Dual timestamping enables a database to capture both the enterprise history and its sequence of changes, providing temporal context for the data. It permits applications to query and modify data using temporal dimensions without losing historical data (Torp et al. 2000). Stratum architecture, for instance, enables the evaluation of different timestamping approaches for bitemporal data. This architecture allows testing and comparison of different methods to better understand their performance under varying conditions and supports the management of historical and projected data (Torp et al. 1998).

Bitemporal capacity for historical preservation directly affects query processing and complexity, demonstrating how bitemporal indexing reduces the computational overhead of temporal retrieval. For example, Salzberg and Tsotras (1999) compare access methods for time-evolving data, while Kumar et al. (1998) propose indexing strategies that bridge theoretical models and practical query performance. Kaufmann et al. (2015) expand on this with the Bitemporal Timeline Index, modelling valid and transaction time as independent indexes, outperforming traditional structures in selection, join, and aggregation. For instance, BiSW (Bitemporal Sliding Window) is a two-dimensional index designed for bitemporal windows and can process queries over historical and streaming data. This solution addresses the challenges of continuously arriving data alongside older, static records and query complexity by ensuring fast lookup speed while minimising maintenance overhead and space complexity when processing both fixed and sliding bitemporal window queries, making query processing faster and less resource intensive (Ge 2014).

Closely related to these query properties, the other advantage is security and auditability. Sedighi et al. (2020) demonstrate how dual-temporal tracking ensures compliance-grade audit trails in financial services. Meanrach and Chittayasothorn (2007) discuss these features in their paper by integrating valid time and transaction time with multilevel security mechanisms that enable databases to accurately record when information is considered true and when it is stored across multiple clearance levels. This shows that, with strict security controls, knowledge of the past, present, and future, along with record timestamps, supports accurate auditing and historical traceability across security levels.Furthermore, bitemporal Systems can be easily incorporated into existing Database Management Systems (DBMS) without incurring high additional cost for enterprises seeking to leverage temporal data Management capabilities. For instance, in the XBiT model, temporal data are represented using XML, enabling compact, intuitive storage of temporal history. This structure is more accessible to end users and more cost-efficient, since it can be applied to existing systems without major changes to the initial infrastructure (Tang and Tang 2007).

Overall, these advantages are not limited to individual studies but are consistently observed across modelling, indexing, and system-level research within the reviewed literature.

7 Research gap and future research directions

Despite the functional strengths discussed above, several important challenges and underexplored areas remain in bitemporal database research. The following section, therefore, examines these gaps and outlines key directions for future research.

The findings of this review indicate that cloud computing is becoming a trend in storage. For example, this paper discusses how Cloud computing assumes a leading role by providing services and resources, ranging from network and communication management to data storage and processing (Sedighi et al. 2020). However, it provides elastic storage; existing approaches do not adequately optimise bitemporal storage and query processing. This shows that more research is needed on cloud-based bitemporal storage areas (Al-Romema and Nashwan 2010). A key technical challenge in this context is maintaining temporal consistency guarantees for retroactive updates in eventually consistent distributed cloud architectures, where the strong consistency assumed by bitemporal semantics conflicts directly with the CAP theorem trade-offs typical of cloud-native NoSQL systems (Eshtay et al. 2019).

The reviewed literature also highlights the need for improved methods for visualising and representing bitemporal data in both 2D and 3D environments to determine the complexity of data flows in temporal databases. For example, valid time and transaction events can be created for the registration of the plan of survey, thereby improving database accuracy. However, to make it available to a wider range of users, we need sophisticated software that can distinguish between real-world database changes (Thompson and Oosterom 2021).

The review also identifies recent developments in graph-based bitemporal systems, particularly the bitemporal property graph model TPGM+, developed through collaboration between Oracle Corporation and the University of Leipzig, it introduced T-PGQL, a declarative graph query language for matching patterns in a TPGM+graph, and an event detection engine to register T-PGQL queries on bitemporal graph storage for continuous graph change notifications, which stores TPGM and graphs in temporal relational tables with modify/query functionality and registers queries for event detection (Christy and Meeragandhi 2015). Even with all these improvements, there are still a few challenges related to high data velocity, heterogeneous node and edge types, and real-time analytics requirements, which require new ways to store, query, and analyse bitemporal graph data. Addressing these challenges opens significant research potential to extend the bitemporal property graph application beyond the use-case paradigm (Rost et al. 2021).

Perhaps the most significant open direction identified in this review concerns the integration of bitemporal database structures with machine learning, an area that remains immature despite growing practical demand. For example, this paper presents a method for handling multiple primitive events that occur in parallel or sequentially over time (Christy and Meeragandhi 2015). The authors have defined a framework that combines a Bayesian network with the interval algebra to model temporal dependencies and have introduced advanced machine learning methods to learn the model structure and parameters over a time period that includes historical data (Zhang et al. 2013).

Overall, these research gaps indicate that while bitemporal database research has matured conceptually, its practical realisation in scalable, integrated, and application-driven systems remains an open challenge.

7.1 Limitations

While bitemporal data research shows strong potential, several implementation and scalability challenges remain evident across existing systems. These limitations are not isolated but are consistently reflected across modelling, indexing, and system-level studies, particularly in environments characterised by high data volume and continuous updates.

Scalability is a challenge, especially with high-frequency or high-volume time-stamped data, such as video streaming and real-time transaction processing, both of which generate continuous flows of time-stamped records that push storage and processing resources to their limits. Most bitemporal models were built for more stable, lower-volume workloads, so they tend to fall short when deployed in fast-moving environments where response times must stay low, and throughput must remain high (Sedighi et al. 2020).In transaction-time database systems, updates are non-destructive, as new record versions are inserted while previous states are retained, resulting in an append-only versioning model. This is useful for maintaining history, but the version chains it produces keep growing with every change, causing significant storage and processing overhead, especially in high-update environments where long version chains must be managed efficiently and queried (Lomet et al. 2006).

In addition to scalability concerns, bitemporal databases introduce significant semantic complexity due to the simultaneous management of valid and transaction time. Unlike traditional databases, updates are non-destructive, leading to rapid expansion and large storage needs. The real bottleneck is processing and querying this historical dataset. Consider data sources such as IoT sensor readings, financial market data feeds, or real-time logistics tracking, which may generate y large volumes of data at high velocity. Existing bitemporal models and their underlying data structures are often not optimised for the computational overhead of efficiently querying such massive datasets (Kvet 2025). In IoT streaming environments specifically, sensor data with irregular sampling rates and delayed transmission creates overlapping valid-time windows that are difficult to reconcile in a bitemporal schema, particularly when the sensor's reported timestamp diverges significantly from the database recording timestamp due to network latency or device failure (Kudo 2019).

Thus, bitemporal databases have a few limitations, but their benefits tend to outweigh these challenges. For instance, recent advancements, such as the SQL:2023 standard, provide true support for multi-dimensional arrays. These are useful for analytics and time-series data, where changes occur across multiple temporal dimensions, and they enable future business integration (Misev and Baumann 2015). Furthermore, ACID (Atomicity, Consistency, Isolation, and Durability) principles, which are fundamental to database transactions, can be applied to bitemporal databases to ensure consistent data management across multiple time dimensions. This demonstrates that bitemporality supports more informed decision-making in a dynamic business environment (Wondoh et al. 2015).

8 Conclusions

This study enhances the literature by integrating previous research and providing a comprehensive overview of bitemporal database systems. It employs a PRISMA-guided selection process, complemented by bibliometric and thematic analyses.

Firstly, the study reveals key challenges inherent in implementing a bitemporal system, including indexing strategies, schema version management, graph-based extensions, and inconsistent DBMS support. Although SQL:2011 introduced temporal features, full support for bitemporal functionality is still not consistently implemented across database systems. Many solutions continue to depend on custom modelling or architectural extensions.

Secondly, this research maps the current research terrain, identifies key contributors and significant publications, and develops thematic trends in the domain. This approach enhances transparency and reproducibility in the review methodology. Furthermore, it addressed critical gaps, including limited DBMS-level support, insufficient research on combining bitemporal models with machine learning and predictive analytics, the need for more realistic industrial datasets to capture practical real-world challenges, and the need for a lighter, more flexible bitemporal architecture for an evaluation framework to check correctness, system performance, and the reproducibility of temporal results.

Thirdly, despite existing gaps, the study highlights the practical and theoretical contributions of the bitemporal database for managing evolving data, particularly in improving temporal representation accuracy, data availability, and informed decision-making. Finally, the study shows that the bitemporal approach addresses the key limitations of traditional systems for handling historical data by storing both valid and transaction times. This dual-timeline approach offers better data integrity and traceability.

To conclude, the strict inclusion criteria and the focus on bitemporal-related papers may have overlooked relevant work from other contexts, particularly in non-English-language publications. From an applied perspective, these findings offer a useful map for organisations and system designers evaluating the advantages and implementation obstacles of adopting bitemporal databases, as well as future pathways for the development and application of bitemporal database systems across domains.

Appendix 1

No	References	Q1	Q2	Q3	Q4	Qscores
1	Kvet et al. 2014a)	1	1	1	1	4
2	Tang et al. 2004)	1	0.5	0.5	1	3
3	Salzberg and Tsotras 1999)	1	0.5	1	1	3.5
4	Johnston 2014)	1	1	1	1	4
5	Johnston and Weis 2010)	1	1	1	1	4
6	Kaufmann et al. 2014)	1	1	1	1	4
7	Tansel et al. 2025)	0.5	1	0.5	1	3
8	Chau and Chittayasothorn 2021)	1	0.5	0.5	1	3
9	Cappelli et al. 1995)	1	1	1	1	4
10	Davies et al. 1995)	1	0.5	1	1	3.5
11	Mareco and Bertossi 1999)	1	1	1	1	4
12	Dyreson 1995)	0.5	1	1	1	3.5
13	Bohlen et al. 2000)	1	1	1	1	4
14	Du and Zhu 2024)	0.5	1	1	1	3.5
15	Jensen and Snodgrass 1996)	1	0.5	0.5	1	3
16	Snodgrass 1999)	1	1	1	1	4
17	Jensen et al. 1994)	1	1	1	1	4
18	Kumar et al. 1998)	0.5	1	0.5	1	3
19	Wondoh et al. 2016)	1	0.5	0.5	1	3
20	Sedighi et al. 2020)	1	1	1	1	4
21	Chen et al. 2024)	1	0.5	1	1	3.5
22	Atay and Alp 2016)	1	1	1	1	4
23	Wang and Zaniolo 2004)	1	0.5	0.5	1	3
24	Kaufmann et al. 2015)	1	1	1	1	4
25	Zhang et al. 2024)	1	0.5	1	1	3.5
26	Atay and Tansel 2009)	1	1	1	1	4
27	Njovu and Ibrahim 2003)	0.5	1	1	1	3.5
28	El Hayat et al. 2020)	0.5	1	1	1	3.5
29	Stantic et al. 2009)	1	1	1	1	4
30	Nascimento et al. 1996)	0.5	1	1	1	3.5
31	Agesen et al. 2001)	1	0.5	0.5	1	3
32	Eid 2002)	1	1	1	1	4
33	Stantic et al. 2004)	1	1	1	1	4
34	Saltenis and Jensen 2002)	1	1	1	1	4
35	Wei and Elmasri 2000)	1	1	1	1	4
36	Brahmia and Bouaziz 2008)	1	0.5	0.5	1	3
37	Rost et al. 2111)	1	1	1	1	4
38	Saroha and Gosain 2016)	1	0.5	1	1	3.5
39	Petkovi´c 2017)	1	1	1	1	4
40	Eshtay et al. 2019)	1	1	1	1	4
41	Rost et al. 2022)	1	1	1	1	4
42	Gancarski 1999)	1	1	1	1	4
43	Hou et al. 2023)	1	0.5	0.5	1	3
44	Sedighi et al. 2020)	1	1	1	1	4
45	Peng et al. 2020)	1	1	1	1	4
46	Anselma et al. 2010)	0.5	1	0.5	1	3
47	Torp et al. 2000)	1	0.5	0.5	1	3
48	Torp et al. 1998)	1	1	1	1	4
49	Knolmayer and Myrach 2001)	1	0.5	1	0.5	3
50	Ge 2014)	0.5	1	1	0.5	3
51	Tang and Tang 2007)	1	0.5	1	0.5	3
52	Al-Romema and Nashwan 2010)	1	0.5	1	1	3.5
53	Thompson and Oosterom 2021)	1	1	1	0.5	3.5
54	Wondoh et al. 2015)	1	1	1	1	4

Appendix 2

Models	Application sector	Time stamping technique	Bitemporal support	Query language used	Limitations
BOSEMA (Njovu and Ibrahim 2003)	Decision Support Systems	Object-level timestamping	Yes	Design-pattern-based	No indexing or query optimisation focus
UML/OCL–TORDB (El Hayat et al. 2020)	Enterprise Information Systems	Attribute-level bitemporal timestamping	Yes	SQL + OCL constraints	Does not address query performance or indexing
POINT Approach (Stantic et al. 2009)	General-purpose relational DB	Point-based timestamp representation	Yes	SQL	Mainly optimises NOW queries; limited semantic enrichment
Bi-temporal Timeline Index (Nascimento et al. 1996)	Database Systems	Independent 1D timeline checkpointing	Yes	SQL	Checkpoint maintenance overhead
Split Operator (Now-relative) (Agesen et al. 2001)	Relational DB Systems	Point-based segmentation	Yes	SQL	Increased complexity in temporal splitting
B-tree (Baseline Temporal Index) (Eid 2002)	General DB Systems	One-dimensional ordering	Partial	SQL	Inefficient for interval/multi-dimensional temporal queries
4-R Index (Stantic et al. 2004)	Temporal DB Systems	Bitemporal function mapping to fixed regions	Yes	SQL	Specialized for specific workload types
HR-tree (Saltenis and Jensen 2002)	Historical DB Systems	Version-based hierarchical indexing	Yes	SQL	Storage overhead due to multiple versions
Schema Versioning Model (Wei and Elmasri 2000)	Enterprise Databases	Schema-level version timestamping	Yes	SQL	Focuses more on schema evolution than runtime optimisation
Multi-temporal XML Model (Brahmia and Bouaziz 2008)	Web/XML Systems	Document-level timestamping	Yes	XQuery + SQL	Heavy & complex for large XML repositories
TPGM + (Bitemporal Property Graph Model) (Rost et al. 2021)	IoT, Knowledge Graphs, Digital Twins	Node/Edge timestamping	Yes	T-PGQL	High complexity & learning curve

Author contributions All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by Gokul Neelamegam. The first draft of the manuscript was written by Gokul Neelamegam, and all authors commented on earlier versions. All authors read and approved the final manuscript.

Funding The research was conducted as part of a doctoral program at Birmingham City University. No specific external funding was received for this study.

Data availability This study is based on a PRISMA-guided systematic literature review of publicly available peer-reviewed publications. The data supporting the findings consist of extracted bibliographic information, quality assessment results, and thematic classifications derived from the selected studies. All analysed data are presented within the manuscript.

Declarations

Ethical approval and consent to participate This study is based on a systematic literature review and bibliometric analysis of publicly available academic publications. It did not involve human participants or animal subjects. Therefore, ethical approval was not required.

Competing interests The authors declare no competing interests.

References

Adnan K, Akbar R, Wang KS (2021) Development of usability enhancement model for unstructured big data using SLR. IEEE Access 9:87391–87409. https://doi.org/10.1109/ACCESS.2021.3089100

Agesen M, B¨ohlen MH, Poulsen LO, Torp K (2001) A split operator for now relative bitemporal databases. In: Proceedings of the

17th IEEE International Conference on Data Engineering (ICDE 2001), pp 41–50. https://doi.org/10.1109/ICDE.2001.914812

Al-Romema N, Nashwan N (2010) Memory storage issues of temporal database applications on relational database management systems. J Comput Sci 6(3):296–304. https://doi.org/10.3844/jcssp.2010.296.304

Anselma L, Terenziani P, Snodgrass RT (2010) Valid-time indeterminacy in temporal relational databases: A family of data models. In: 17th International Symposium on Temporal Representation and Reasoning (TIME 2010), pp 139–145. https://doi.org/10.1109/TIME.2010.15

Asadi S, Abdullah R, Yah Y, Nazir S (2019) Understanding institutional repository in higher learning institutions: a systematic literature review and directions for future research. IEEE Access 7:35242–35263

Asghar MZ, Alam KA, Javed S (2019) Software design patterns recommendation : A systematic literature review. In: 2019 International Conference on Frontiers of Information Technology (FIT), pp 167–172. https://doi.org/10.1109/FIT47737.2019.00040

Atay CE, Alp G (2016) Modeling and querying multidimensional bitemporal data warehouses. Int J Comput Comm Eng 5(2):110–119. https://doi.org/10.17706/IJCCE.2016.5.2.110-119

Atay CE, Tansel AU (2009) Bitemporal Databases: Modeling and Implementation. VDM Verlag, Saarbru¨cken, Germany

Bliujute R, Jensen CS, Saltenis S, Slivinskas G (2000) Light-weight indexing of general bitemporal data. In: Proceedings. 12th International Conference on Scientific and Statistica Database Management, pp 125–138. https://doi.org/10.1109/SSDM.2000.869783

B¨ohlen MH, Dign¨os A, Gamper J, Jensen CS (2018a) Temporal data management: An overview. In: Zim´anyi E (ed) Business Intelligence and Big Data. Springer, Cham, pp 51–83. https://doi.org/10.1007/978-3-319-96655-73

B¨ohlen MH, Dign¨os A, Gamper J, Jensen CS (2018b) Database technology for processing temporal data. In: Proceedings of the 25th International Symposium on Temporal Representation and Reasoning (TIME 2018), pp 2–127. https://doi.org/10.4230/LIPIcs.TIME.2018.2

Bohlen MH, Jensen CS, Snodgrass RT (2000) Temporal statement modifiers. ACM Trans Database Syst 25(4):407–456. https://doi.org/10.1145/377674.377665

Brahmia Z, Bouaziz R (2008) Schema versioning in multi-temporal xml databases. In: Seventh IEEE/ACIS International Conference on Computer and Information Science (ICIS 2008), pp 158–164. https://doi.org/10.1109/ICIS.2008.89

Brahmia Z, Grandi F, Bouaziz R (2023) Temporal blockchains for intelligent transportation management and autonomous vehicle support in the internet of vehicles. In: Modelling and Simulation of Fast-Moving Ad-Hoc Networks (FANETs and VANETs), pp 155–189. https://doi.org/10.4018/978-1-6684-3610-3.ch009

Cappelli A, De Castro C, Scalas MR (1995) A modular history-oriented access structure for bitemporal relational databases. In: SOFSEM '95: Theory and Practice of Informatics. Lecture Notes in Computer Science, vol 1012, pp 246–255. https://doi.org/10.1007/3-540-60609-219

Chau VTN, Chittayasothorn S (2021) A bitemporal sql database design method from the enhanced entity-relationship model. In: 2021 7th International Conference on Engineering, Applied Sciences and Technology (ICEAST), pp 85–90. https://doi.org/10.1109/ICEAST52143.2021.9426270

Chen K, Liu C, Li W, Liu Z, Chen H, Zhang H, Zou Z, Shi Z (2024) Time travelling pixels: Bitemporal features integration with foundation model for remote sensing image change detection. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2024), pp 8581–8584. https://doi.org/10.1109/IGARSS53475.2024.10640593

Christy A, Meeragandhi G (2015) Combining bitemporal conceptual data model with multiway join relations for forecasting. Proc Comput Sci 57:1104–1114. https://doi.org/10.1016/j.procs.2015.07.396

Davies C, Lazell B, Hughes M, Cooper L (1995) Time is just another attribute, or at least, just another dimension. In: Clifford J, Tuzhilin A (eds) Recent Advances in Temporal Databases. Springer, London, pp 175–193. https://doi.org/10.1007/978-1-4471-3033-810

Debrouvier A, Perazzo M, Parodi E, Soliani V, Vaisman A (2021) A model and query language for temporal graph databases. VLDB J 30:825–858. https://doi.org/10.1007/s00778-021-00675-4

Ding L, Meng L (2014a) A comparative study of thematic mapping and scientific visualization. Ann GIS 20(1):23–37. https://doi.org/10.1080/19475683.2013.862856

Ding L, Meng L (2014b) A comparative study of thematic mapping and scientific visualization. Ann GIS 20(1):23–37. https://doi.org/10.1080/19475683.2013.862856

Donthu N, Kumar S, Mukherjee D, Pandey N, Lim WM (2021) How to conduct a bibliometric analysis: an overview and guidelines. J Bus Res 133:285–296. https://doi.org/10.1016/j.jbusres.2021.04.070

Doulani A (2021) A bibliometric analysis and science mapping of scientific publications of Alzahra University during 1986–2019. Libr Hi Tech 39(4):915–935. https://doi.org/10.1108/LHT-06-2020-0131

Du Y, Zhu L (2024) Digital marketing strategies for e-commerce supply chain risk management based on big data analysis and predictive modeling. ComputerAided Design and Applications, pp 1–18. https://doi.org/10.14733/cadaps.2024.S4.1-18

Dyreson C (1995) The tsql2 temporal query language. In: Snodgrass RT (ed) The TSQL2 Temporal Query Language. Springer, Boston, MA, pp 1–26. https://doi.org/10.1007/978-1-4615-2289-8

Eid A (2002) Efficient associative data structures for bi-temporal databases. PhD thesis, Royal Roads University. https://doi.org/10.20381/ruor-11150

El Hayat SA, Toufik F, Bahaj M (2020) UML/OCL based design and the transition towards temporal object relational database with bitemporal data. J King Saud Univ Comput Inf Sci 32(4):398–407. https://doi.org/10.1016/j.jksuci.2019.08.012

Eshtay M, Sleit A, Aldwairi M (2019) Implementing bi-temporal properties into various NoSQL database categories. Int J Comput 18(1):45–52

Farina M, Gorb A, Kruglov A, Succi G (2022) Technologies for GQM-based metrics recommender systems: a systematic literature review. IEEE Access 10:23098–23111. https://doi.org/10.1109/ACCESS.2022.3152397

Fredrick T, Radhamani G (2015) The fuzzy logic based eca rule processing for xml databases. Int Arab J Inf Technol (IAJIT) 12

Gancarski S (1999) Database versions to represent bitemporal databases. In: Database and Expert Systems Applications (DEXA 1999). Lecture Notes in Computer Science, vol 1677, pp 832–841. https://doi.org/10.1007/3-540-48309-878

Ge C (2014) Bitemporal sliding windows. PhD thesis, University of Waterloo

Hilmi MAA, Puspaningrum A, Darsih Siahaan DO, Samosir HS, Rahma AS (2023) Research trends, detection methods, practices, and challenges in code smell: SLR. IEEE Access 11:129536–129551. https://doi.org/10.1109/ACCESS.2023.333425

Hou J, Zhao Z, Wang Z, Lu W, Jin G, Wen D, Du X (2023) Aeong: An efficient built-in temporal support in graph databases. arXiv preprint arXiv:2304.12212. https://doi.org/10.48550/arXiv.2304.12212

Jensen CS, Snodgrass RT (1996) Semantics of time-varying information. Inf Syst 21(4):311–352. https://doi.org/10.1016/0306-4379(96)00017-8

Jensen CS, Soo MD, Snodgrass RT (1994) Unifying temporal data models via a conceptual model. Inf Syst 19(7):513–547. https://doi.org/10.1016/0306-4379(94)90013-2

Johnston T (2014) Bitemporal Data: Theory and Practice. Morgan Kaufmann, Waltham, MA. https://doi.org/10.1016/C2012-0-06609-4

Johnston T, Weis R (2010) Managing Time in Relational Databases: How to Design, Update and Query Temporal Data. Morgan Kaufmann, Burlington, MA. https://doi.org/10.1016/C2009-0-20326-6

Kaufmann M, Fischer PM, May N, Ge C, Goel AK, Kossmann D (2015) Bitemporal timeline index: A data structure for processing queries on bi-temporal data. In: Proceedings of the 31st IEEE International Conference on Data Engineering (ICDE 2015), pp 471–482. https://doi.org/10.1109/ICDE.2015.7113307

Kaufmann M, Fischer PM, May N, Kossmann D (2014) Benchmarking bitemporal database systems: Ready for the future or stuck in the past? In: Advances in Database Technology – EDBT 2014. Proceedings of the 17th International Conference on Extending Database Technology, pp 738–749. https://doi.org/10.5441/002/edbt.2014.80

Kaur S, Dhindsa KS (2016) Comparative study of citation and reference management tools: Mendeley, Zotero and ReadCube. In: 2016 International Conference on ICT in Business Industry & Government (ICTBIG), pp 1-5. https://doi.org/10.1109/ICTBIG.2016.7892715

Klarin A (2024) How to conduct a bibliometric content analysis: guidelines and contributions of content co-occurrence or co-word literature reviews. Int J Consum Stud 48(2):13031. https://doi.org/10.1111/ijcs13031

Kleiner C, Lipeck UW (2002) Performance of querying temporal attributes in objectrelational databases. In: Proceedings of the Ninth International Symposium on Temporal Representation and Reasoning, pp 58–60. https://doi.org/10.1109/TIME.2002.1027475

Knolmayer GF, Myrach T (2001) Concepts of bitemporal database theory and the evolution of web documents. In: Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS 2001), pp 1–10. https://doi.org/10.1109/HICSS.2001.927091

Kudo T (2019) Data correction management method using temporal data in fog computing. In: Proceedings of the 12th International Conference on Mobile Computing and Ubiquitous Network (ICMU 2019), pp 1–4. https://doi.org/10.23919/ICMU.2019.8874094

Kulkarni K, Michels J-E (2012) Temporal features in SQL:2011. ACM SIGMOD Rec 41(3):34–43. https://doi.org/10.1145/2380776.2380786

Kumar A, Tsotras VJ, Faloutsos C (1998) Designing access methods for bitemporal databases. IEEE Trans Knowl Data Eng 10(1):1–20. https://doi.org/10.1109/69.667079

Kvet M (2025) Temporal constraint modeling. In: 2025 Communication and Information Technologies (KIT), pp 1–8. https://doi.org/10.1109/KIT67756.2025.11205437

Kvet M, Matiako K, Kvet M (2014a) Transaction management in fully temporal system. In: 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp 148–153. https://doi.org/10.1109/UKSim.2014.26

Kvet M, Matiaˇsko K, Kvet M (2014b) Complex time management in databases. Cent Eur J Comput Sci 4:269–284. https://doi.org/10.2478/s13537-014-0207-4

Lomet D, Barga R, Mokbel MF, Shegalov G (2006) Transaction time support inside a database engine. In: 22nd International Conference on Data Engineering (ICDE'06), pp 35–35. https://doi.org/10.1109/ICDE.2006.162

Mareco CA, Bertossi L (1999) Specification and implementation of temporal databases in a bitemporal event calculus. In: Conceptual Modeling – ER 1999. Lecture Notes in Computer Science, vol 1728, pp 74–85. https://doi.org/10.1007/3-540-48054-47

Meanrach W, Chittayasothorn S (2007) A bitemporal multilevel secure database system. In: AFRICON 2007, pp 1–7. https://doi.org/10.1109/AFRCON.2007.4401491

Misev D, Baumann P (2015) SQL support for multidimensional arrays: extending standard relational databases with arrays. Proc VLDB Endow 8(12):1922–1933. https://doi.org/10.14778/2824032.2824071

Moumen A, Bouchama EH, El Bouzekri El Idirissi Y (2020) Data mining techniques for employability: Systematic literature review. In: Proceedings of the 2nd IEEE International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS 2020), pp 1–5. https://doi.org/10.1109/ICECOCS50124.2020.9314555

Nascimento MA, Dunham MH, Elmasri R (1996) M-ivtt: An index for bitemporal databases. In: Database and Expert Systems Applications (DEXA 1996). Lecture Notes in Computer Science, vol 1134, pp 779–790. https://doi.org/10.1007/BFb0034730

Njovu C, Ibrahim MT (2003) A design pattern approach to bitemporal data modeling. In: Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DEXA 2003). IEEE, pp 758–761. https://doi.org/10.1109/DEXA.2003.1232112

Peng X, Long G, Shen T, Wang S, Jiang J, Zhang C (2020) Bitenet: Bidirectional temporal encoder network to predict medical outcomes. In: 2020 IEEE International Conference on Data Mining (ICDM), pp 412–421. https://doi.org/10.1109/ICDM50108.2020.00050

Petkovi´c D (2017) Temporal data in relational database systems: A comparison. In: Handbook of Research on Advanced Trends in Database Technologies and Management, pp 18–27. https://doi.org/10.1007/978-3-319-31232-32

Rost C, Fritzsche P, Schons L, Zimmer M, Gawlick D, Rahm E (2021) Bitemporal property graphs to organize evolving systems. arXiv preprint arXiv:2111.13499. https://doi.org/10.48550/arXiv.2111.13499

Rost C, Gomez K, T¨aschner M, Fritzsche P, Schons L, Christ L, Adameit T, Junghanns M, Rahm E (2022) Distributed temporal graph analytics with Gradoop. VLDB J 31(2):375–401. https://doi.org/10.1007/s00778-021-00687-4

Saltenis S, Jensen CS (2002) Indexing of now-relative spatio-bitemporal data. VLDB J 11:1–16. https://doi.org/10.1007/s007780100058

Salzberg B, Tsotras VJ (1999) Comparison of access methods for time-evolving data. ACM Comput Surv 31(2):158–221. https://doi.org/10.1145/319806.319816

Saroha K, Gosain A (2016) Bi-temporal schema versioning in bi-temporal data warehouse. CSI Trans ICT 3(2–4):135–142. https://doi.org/10.1007/s40012-016-0081-4

Sedighi A, D'Mello S, Isaacs G, Jacobson D (2020) Bitemporaldb: A bitemporal database in the cloud for financial services data. In: 2020 IEEE International Conference on Smart Cloud (SmartCloud), pp 68–73. https://doi.org/10.1109/SmartCloud49737.2020.00021

Singh G, Chaturvedi AK (2021) Particle swarm optimization-based approaches for cloud-based task and workflow scheduling: A systematic literature review. In: Proceedings of the 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC 2021), pp 350–358. https://doi.org/10.1109/ICSCCC51823.2021.9478149

Snodgrass RT (1990) Temporal databases: status and research directions. ACM SIGMOD Rec 19(4):83–89. https://doi.org/10.1145/122058.122068

Snodgrass RT (1999) Developing Time-oriented Database Applications in SQL. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA

Stantic B, Khanna S, Thornton J (2004) An efficient method for indexing nowrelative bitemporal data. In: Proceedings of the Fifteenth Australasian Database Conference (ADC 2004), pp 113–122

Stantic B, Sattar A, Terenziani P (2009) The point approach to represent now in bitemporal databases. J Intell Inf Syst 32(3):297–323. https://doi.org/10.1007/s10844-008-0072-5

Tang N, Tang Y (2007) Mapping bitemporal xml data model to xml document. In: Computer Supported Cooperative Work in Design. Springer, pp 342–352. https://doi.org/10.1007/978-3-540-92719-831

Tang Y, Liang L, Huang R, Yu Y (2004) Bitemporal extensions to non-temporal rdbms in distributed environments. In: 8th International Conference on Computer Supported Cooperative Work in Design, vol 2, pp 370–3732. https://doi.org/10.1109/CACWD.2004.1349216

Tansel AU, Wu D, Wang H-T (2025) Time travel with the bitemporal RDF model. Mathematics 13(13):2109. https://doi.org/10.3390/math13132109

Taylor KS, Mahtani KR, Aronson JK (2021) Summarising good practice guidelines for data extraction for systematic reviews and meta-analysis. BMJ Evid Based Med 26(3):88–90. https://doi.org/10.1136/bmjebm-2020-111651

Thompson R, Oosterom P (2021) Bi-temporal foundation for LADM v2: fusing eventand state-based modelling of land administration data 2D and 3D. Land Use Policy 102:105246. https://doi.org/10.1016/j.landusepol.2020.105246

Torp K, Jensen CS, Snodgrass RT (1998) Stratum approaches to temporal dbms implementation. In: Proceedings of the International Database Engineering and Applications Symposium (IDEAS 1998), pp 4–13. https://doi.org/10.1109/IDEAS.1998.694346

Torp K, Jensen CS, Snodgrass RT (2000) Effective timestamping in databases. VLDB J 8(3):267–288. https://doi.org/10.1007/s007780050008

Trudel A, Goodwin SD (2000) Proceedings of the international workshop on temporal representation and reasoning. In: Proceedings of the 7th International Workshop on Temporal Representation and Reasoning (TIME 2000), pp 133–141. https://doi.org/10.1109/TIME.2000.856594

Wang F, Zaniolo C (2004) Xbit: An xml-based bitemporal data model. In: Proceedings of the International Conference on Conceptual Modelling, pp 810–824. https://doi.org/10.1007/978-3-540-30464-760

Wang H-T, Tansel AU (2019) Temporal extensions to RDF. J Web Eng 18(1–3):125–168. https://doi.org/10.13052/jwe1540-9589.18134

Wang Q-G, Li X-L, Wang S-W (2013) A dynamic fuzzy cluster algorithm for time series. Sci Program. https://doi.org/10.1155/2013/183410

Wei H-C, Elmasri R (2000) Schema versioning and database conversion techniques for bi-temporal databases. Ann Math Artif Intell 30(1):23–52. https://doi.org/10.1023/A:1016622202755

Wondoh J, Grossmann G, Gasevic D, Reichert M, Schrefl M, Stumptner M (2015) Bitemporal support for business process contingency management. In: International Conference on Conceptual Modeling, pp 109–118. https://doi.org/10.1007/978-3-319-25747-111

Wondoh J, Grossmann G, Stumptner M (2016) Utilising bitemporal information for business process contingency management. In: Proceedings of the Australasian Computer Science Week Multiconference (ACSW 2016). ACM, pp 1–10. https://doi.org/10.1145/2843043.2843045

Won J, Elmasri R (1996) Representing retroactive and proactive versions in bitemporal databases (2tdb). In: Proceedings of the Twelfth International Conference on Data Engineering (ICDE 1996), pp 85–94. https://doi.org/10.1109/ICDE.1996.492092

Zhang F, Zhang W, Wang G (2024) A bitemporal RDF index based on skip list. Intell Data Anal 28(6):1579–1599. https://doi.org/10.3233/IDA-230609

Zhang Y, Zhang Y, Swears E, Larios N, Wang Z, Ji Q (2013) Modeling temporal interactions with interval temporal Bayesian networks for complex activity recognition. IEEE Trans Pattern Anal Mach Intell 35(10):2468–2483. https://doi.org/10.1109/TPAMI.2013.33