Date available online: 30-10-2023
Vol. 1, Issue 1(January-June 2023)
This is an open-access article.

Data Models, Semantics, Query languages

Rameesha Ahmad[1],Muhammad Zunnurain Hussain[2*], Muhammad Zulkifl Hasan[3], Muhammad Abubakar Afaq[1].

[1]Department of Economics with Data Science, Information Technology University
[2] Department of Computer Science Bahria University Lahore Campus
[3]Faculty of Information Technology, University of Central Punjab, Lahore Pakistan

[*]Corresponding author: Muhammad Zunnurain Hussain (e-mail: zunnurain.bulc@bahria.edu.pk).

ABSTRACT This study tries to offer a complete review of database concepts, semantics, and query languages. A data model is a theoretical data representation that specifies the structure, connections, and restrictions of data in a certain area. It allows you to organize and categorize data for easier organization, access, and management. A query language is a kind of computer programming language that is used to get information from a database. It enables users to describe the data they wish to obtain as well as the criteria under which the data must be provided. In the overall context of data modeling and query languages, semantics refers to the interpretation and significance of data and questions. It entails setting the rules and limitations for data storage, retrieval, and manipulation, as well as verifying that query results are consistent and relevant. Data modeling, query language, and semantics work together to create the basis of database management systems, allowing for efficient and effective data storage and retrieval for a broad variety of applications. We intend to give a greater knowledge of these ideas and their usefulness in data management by exploring the strengths and shortcomings of various data models and query languages, as well as highlighting the problems and applications of semantics. Here, also determines the expansion for the global usage of EHR systems and the demand for efficient data management services which are built for the betterment of health records. A precise literature study was carried out, encompassing a wide variety of publications and themes such as Data Models, Semantics, and Query Languages, the importance of these ideas in database systems, and an assessment of the advantages and drawbacks of numerous data models.

Index Terms**—***data structures, semantics, query languages, database systems, data models, programming languages, strengths, drawbacks, syntax, benefits, contrast, analyses, semantics, practice, electronic health records.*

I. INTRODUCTION

This article will go through data structures, semantics, and query languages in detail. We'll start by going over each topic and discussing why it's important in database systems. We will then analyze and contrast several data models, presenting examples of the way each model is utilized in practice and assessing each model's strengths and drawbacks. We will also go over the many forms of semantics that are used in programming languages and databases, as well as provide instances of how semantics are used in practice and examine the issues that come with applying semantics to database systems. Finally, we will contrast the two query languages, analyzing their syntax, semantics, and benefits and drawbacks.

A. Data Models

Databases are commonplace in today's society, with massive volumes of data being gathered and kept in industries such as business, healthcare, education, and others. Organizations must have reliable and successful data management systems in place in order to make accurate choices and obtain knowledge about their operations. Database systems rely on three important principles to achieve this goal: data models, semantics, and query languages.

Data models consist of three components, which are:

•a variety of data structure forms such as the foundation of any database based on the structure model.

•a set of operators or information-processing rules that may be used to any credible instance of the data types stated in the preceding paragraph in order to obtain or generate data from any component of such structures in any combination required

•a group of generic integrity rules that specify the series of consistent database states, changes of state, or both – these rules are commonly represented as insert-update-delete rules.

Data models give a systematic approach of expressing and organizing data, allowing for effective data storage, retrieval, and manipulation. There are several sorts of data models, each one having its own set of advantages and disadvantages. The relational model, hierarchical model, network model, and object-oriented model are the most common data models. Having the appropriate data model is crucial for creating an efficient database system.

B. Semantics

Semantics, on the contrary, is associated with the interpretation and meaning of facts. Semantics are vital in database systems because they guarantee that queries and data operations are understandable and correct. Semantics can be classified as a combination of formal, informal, functioning, denotational, and axiomatic semantics. Using semantics in database systems may be difficult, especially when interacting with inaccurate or incomplete information.

C. Query Language

Query languages allow you to access and manipulate information stored in a database. SQL (Structured Query Language) is the most often used query language for retrieving and manipulating data in relational databases. Different query languages utilized for querying various kinds of databases such as XML, RDF, and graph databases include XQuery, SPARQL, and Cypher. Query languages are vital for analyzing and making decisions based on data because they help make sense of massive volumes of data saved in databases.

D. Applications

In many applications, data modeling, semantics, and query languages are employed. Databases are used in the e-commerce sector for keeping information about items, transactions, and consumers. They assist organizations with inventory management, sales tracking, and consumer behavior analysis. Databases are used in the healthcare business to store and handle information about patients, health histories, and research data. They assist healthcare practitioners in improving patient care and furthering medical research. Databases are used in the finance sector to store and manage financial data such as transactions, stocks, and bonds. They assist financial experts in following market movements and making sound investment selections. Databases are used in scientific research to store and organize data from tests and simulations. They assist researchers in analyzing data and making new discoveries.

II. LITERATURE REVIEW

With the increasing number of data-driven software and services over the past few years, the field of database systems has experienced remarkable growth. Effective data management is now more important than ever because of this. The core elements of contemporary database systems are data models, semantics, and query languages. They are essential for organizing data and enhancing performance. We will go further into these subjects in this literature review, talking about the many data models, different kinds of semantics, difficulties in implementing semantics in databases, and comparison and optimization of query languages. The practical application of semantics, the idea of the semantic web, and the significance of database management will also be covered.

A data model is a set of ideas or notations that are used to define data, data connections, semantics, and restrictions. In database design, data models are utilized to organize and arrange data. Data models are classified into three types: conceptual, logical, and physical. Conceptual data models depict the overall structure of the database, whereas logical data models concentrate on the connections between elements. Physical data models define how data is physically stored [1]. In practice, data models such as relational, hierarchical, network, object-oriented, and document-oriented models are utilized. Most database systems employ relational data models, which are the most prevalent. In mainframe systems, hierarchical data models are employed, although network data models are used in some older systems. In object-oriented programming languages, object-oriented data models are utilized, whereas document-oriented models serve the purpose in NoSQL databases [4].

At a high level, the major hurdles are ensuring that the data corresponds with the actual world and that it can be weaved into current business processes and analytics programs. Business managers usually underestimate the amount of effort it takes to clean data [11].

The meaning underlying the symbols and phrases utilized by programming languages and databases is known as semantics. Semantics is classified into three types: denotational, operational, and axiomatic. Denotational semantics is concerned with the significance of expressions, whereas operational semantics is concerned with how statements are assessed. The logical connections between expressions are the subject of axiomatic semantics [2].

There are challenges that are faced in semantics. Regardless of the complexities of natural language and the uncertainty of meaning, implementing semantics to database systems may be difficult. One difficulty is figuring out the meaning of words in the database. Dealing with the many sorts of semantics used in programming languages and databases is another problem. A third problem is ensuring that the database's semantics are consistent and correct [5].

In practice, semantics is employed to verify that data is logical and correct. The usage of ontologies in semantic web applications is one example. Ontologies may be used to guarantee that data is accurately sorted and connected by describing the relationships between ideas [7].

The semantic web is an idealized version of the web's future in which data is connected and readily shared and reused. The semantic web is built on ontologies and linked data. A few instances of semantic web applications include RDF, OWL, and SPARQL frameworks. [9].

To access data from databases, query languages are utilized. SQL, NoSQL, and SPARQL are just a few examples of query languages. SQL is a query language that is often used in relational databases, whereas NoSQL is utilized in non-relational databases. SPARQL is a query language used in semantic web applications [3].

SQL is the most commonly implemented query language, and it is supported by the majority of relational database systems. The query languages used in NoSQL vary greatly based on the sort of database utilized. SPARQL is a query language for RDF data that is used in semantic web applications [6].

The research procedure used in the research project was designed as a three-by-three factorial design. The kind of data model utilized to show database information was one factor. In the experiment, three kinds of data models with different levels of data semantics were adjusted. The second aspect was the SQL's conceptual model type. To incorporate the SQL logical reasoning properties, three kinds of conceptual structures were created. These models of concept were utilized to construct the SQL system's mental representations of the users [26].

Human aspects work in database usage involves both the development of the user administration of dynamic database applications and the design of the database system's online interface. In order to investigate the use of this form of engagement in complicated database contexts demanding a high degree of abstraction, an appropriate data model must be chosen. The semantics of directly acting on a graphical database interface are outlined, potential ambiguities are high- lighted, and a method to set up the user interface in an existing data-base environment is proposed [27].

III. SYSTEMATIC LITERATURE

Reference	Title	Authors	Publication Year	Objective/Research Question	Data Models/Technologies	Key Findings
[1]	Fundamentals of Database Systems	Elmasri, R., Navathe, S. B.	2019	Database system fundamentals	Database fundamentals	Textbook on database systems
[2]	The Formal Semantics of Programming Languages: An Introduction	Winskel, G.	1993	Formal semantics of programming languages	Formal semantics	Introduction to formal semantics
[3]	Scalable Semantic Web Data Management Using Vertical Partitioning	Abadi, D. J., Marcus, A. R., Madden, S., Hollenbach, K.	2013	Semantic web data management	Vertical partitioning	Scalable data management in semantic web
[4]	An Introduction to Database Systems	Date, C. J.	2012	Introduction to database systems	Relational data model	Introduction to database systems
[5]	Semantic Web and Agent Technology: Towards a New Synthesis	Dignum, V.	2011	Semantic web and agent technology	Semantic web technologies	Integration of semantic web and agent technology
[6]	SPARQL Web Query Language	Hartig, O., Bizer, C.	2011	SPARQL web query language	SPARQL	Introduction to SPARQL query language
[7]	Linked Data - The Story So Far	Bizer, C., Heath, T., Berners-Lee, T.	2009	Linked data and semantic web	Linked data	Overview of linked data and semantic web
[8]	An Overview of Query Optimization in Relational Systems	Chaudhuri, S.	1998	Query optimization in relational systems	Query optimization	Overview of query optimization in relational systems
[9]	The Semantic Web	Berners-Lee, T., Hendler, J., Lassila, O.	2001	The semantic web	Semantic web technologies	Introduction to the semantic web
[10]	Database Systems: A Practical Approach to Design, Implementation, and Management	Connolly, T., Begg, C.	2014	Database systems practical approach	Relational data model	Practical approach to database systems
[11]	How to Navigate the Challenges of the Data Modeling Process	Lawton, G.	2019	Data modeling challenges	Data modeling	Article on data modeling challenges
[12]	Fundamentals of Database Systems	Elmasri, R., Navathe, S. B.	2015	Database system fundamentals	Database fundamentals	Textbook on database systems
[13]	An Introduction to Database Systems	Date, C. J.	2013	Introduction to database systems	Relational data model	Introduction to database systems
[14]	A Relational Model of Data for Large Shared Data Banks	Codd, E. F.	1970	Relational model of data	Relational data model	Introduction to relational data model
[15]	Fundamentals of Database Systems	Navathe, S. B., Elmasri, R.	1992	Database system fundamentals	Database fundamentals	Textbook on database systems
[16]	Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics	Sheth, A. P.	1999	Interoperability in information systems	Semantic interoperability	Evolution of interoperability in information systems
[17]	Database System Concepts	Silberschatz, A., Korth, H. F., Sudarshan, S.	2010	Database system concepts	Database fundamentals	Textbook on database systems
[18]	Concurrency Control Performance Modeling: Alternatives and Implications	Agrawal, R., Carey, M. J.	1993	Concurrency control performance modeling	Concurrency control	Alternatives in concurrency control performance modeling
[19]	Concurrency Control Performance Modeling: Alternatives and Implications	Agrawal, R., Carey, M. J.	1993	Concurrency control performance modeling	Concurrency control	Alternatives in concurrency control performance modeling
[20]	Query Optimization in Object-Oriented Database Systems	Chang, C. C., Tung, A. K. H.	1998	Query optimization in object-oriented databases	Query optimization	Query optimization in object-oriented databases
[21]	Logical Foundations of Object-Oriented and Frame-Based Languages	Kifer, M., Lausen, G., Wu, J.	1995	Logical foundations of object-oriented languages	Object-oriented data model	Logical foundations of object-oriented languages

[22]	Performance Modeling of Disk I/O for Database Applications	Li, J., Ross, K. A.	2003	Disk I/O performance modeling	Performance modeling	Disk I/O performance modeling for databases
[23]	The Case for Shared Nothing	Stonebraker, M.	1986	Shared nothing architecture	Shared nothing architecture	Introduction to shared nothing architecture
[24]	Object-Oriented Database Systems: Concepts and Architectures	Zaniolo, C.	1990	Object-oriented database systems	Object-oriented data model	Concepts and architectures of object-oriented databases
[25]	Introduction to Database Systems	Gehani, N.	1990	Introduction to database systems	Database fundamentals	Introduction to database systems
[26]	The Effects of Data Models and Conceptual Models of the Structured Query Language on the Task of Query Writing by End Users	Wu and B. Jiinpo	1991	Effects of data models on query writing	Structured Query Language (SQL)	Effects of data models on query writing
[27]	Implementing Direct Manipulation Query Languages Using an Adequate Data Model	Rainer	1990	Direct manipulation query languages	Adequate data model	Implementing direct manipulation query languages
[28]	Effect of Semantic Distance on Learning Structured Query Language: An Empirical Study	N/A	N/A	N/A	Empirical study on the impact of semantic distance on SQL learning	
[29]	Understanding the Semantic Data Model	N/A	N/A	N/A	Overview of the semantic data model	
[30]	Semantic Modeling - An Overview	Tiwari, S., Jabbar, M. A.	2022	Semantic modeling	Semantic modeling	Overview of semantic modeling
[31]	The Past, Present, and Future of Semantic Search	N/A	N/A	N/A	Overview of the past, present, and future of semantic search	
[32]	9 Data Analytics Trends, Predictions, and Resolutions - 2023	N/A	N/A	N/A	Overview of data analytics trends for 2023	
[33]	Electronic Health Records: Then, Now, and in the Future	Evans, R. S.	2016	Electronic health records	Electronic health records	Evolution of electronic health records
[34]	Security and Privacy in Electronic Health Records: A Systematic Literature Review	Alema´n, J. L. F., Sen˜or, I. C., Lozoya, P. A´. O., Toval, A.	2013	Security and privacy in EHRs	Electronic health records security and privacy	Systematic literature review on EHR security and privacy
[35]	Graph-Representation of Patient Data: A Systematic Literature Review	Schrodt, J., Dudchenko, A., Knaup-Gregori, P., Ganzinger, M.	2020	Graph-representation of patient data	Patient data representation	Systematic literature review on graph-representation of patient data
[36]	E-prescription Rate U.S. 2021	N/A	N/A	N/A	E-prescription adoption rate in the U.S.	
[37]	Ambulatory EHR Market - Growth, Trends, COVID-19 Impact, and Forecasts (2021 - 2026)	N/A	N/A	N/A	Market growth and trends in ambulatory EHR	
[38]	Healthcare Cloud Computing Market Trends, Growth, Drivers & Opportunities	N/A	N/A	N/A	Trends and opportunities in healthcare cloud computing market	

IV. METHODOLOGY

The content, material, and data that have been collected for this report paper are from research articles such as IEEE-explore, Statista, springer, different research papers, etc. to explain the content regarding data models, semantics, and query language. The methodology used for this research is secondary research based on Qualitative Analysis. The conclusions are drawn by examining past literature and the past and current trends of technology. Various papers as mentioned in the literature review Data Models, Semantics, Query languages, benefits, it’s importance in database systems, analysis and contrast of several data models, presenting examples of the way each model is utilized in practice and assessing each model’s strengths and drawbacks and use of Data Models, Semantics, Query languages different disciplinary fields are studied and conclusions are drawn. We will also discuss the many types of semantics used in programming languages and databases, as well as present examples of how semantics are used in practice and evaluate the challenges that arise when applying semantics to database systems. Finally, we will compare the two query languages, examining their syntax, semantics, and advantages and

disadvantages. To achieve our objective, we have extensively reviewed the literature from various credible journal articles and conference papers.
Firstly, we will discuss, what Data Models, Semantics, and Query languages are, and then we will move towards some information about the technology like, how and in what forms is it being used, how it records data, and how is the data stored on it. We will also discuss the development and innovation of Data Models, Semantics, and Query languages and their usage trends in different disciplinary fields. Lastly, we will conclude by discussing the challenges and the future prospects of the Data Models, Semantics, and Query languages.

V. WHAT IS DATA MODEL AND ITS TYPES?

A. Data Model

In order to describe data and its relationships, data models are utilized. They offer a broad overview of the data and its organization. Hierarchical, network, relational, object-oriented, and NoSQL data models are only a few of the numerous categories of data models. Each model has advantages and disadvantages, and the best model to choose depends on the application's needs. The relational model is still frequently used today, although hierarchical and network models are less common. The popularity of object-oriented and NoSQL models has grown as a result of their adaptability and scalability.

B. Hierarchical Data Model

The tree-like structure of data is organized by the hierarchical data model. Each parent record in this model has either one or several child records. The kid records, in return, may have their own child records. When working with data that has a rigid parent-child connection, such as an organizational chart or file system, this approach comes in handy.
The tree-like structure of data is organized by the hierarchical data model. The simplicity of the hierarchical data model is one of its advantages. The format is straightforward to comprehend and browse, making it useful for basic queries. The hierarchical data model, however, has limits. Data having many parent-child connections, for example, or interactions that are not precisely hierarchical, are challenging to manage. It is also difficult to change the layout of the data model after it has been developed.

C. Network Data Model

The network data framework is a hierarchical data model extension that makes it possible for more complicated interactions between records. Records of this type may have many parent and child records. To express the connections between records, the network data model utilizes a set structure. A record type is allocated to each record, and a collection of values is utilized to specify the connections between records of various kinds.
The versatility of the network data structure is one of its advantages. The structure supports complicated record connections, making it appropriate for data with many parent-child relationships. Nevertheless, the network data model is still constrained by its dependence on a fixed structure, which may be difficult to maintain when the data model grows in size. The network data architecture may sometimes be difficult to query since the links between entries are not always obvious.

D. Relational Data Model

When creating a database system, the selection of a data model is crucial. The most popular data model in database systems is the relational data model, which utilizes tables to describe data. However, certain applications still employ different data models, such the hierarchical and network data models. The simplicity and usefulness of the relational data model are two advantages. The application of tables and the SQL query language simplifies data representation and management. Extensive study, testing, and industry approval have also gone into the relational model.
The relational approach, on the other hand, has substantial limitations. Complex relationships between entities may be difficult to explain, and employing joins and subqueries can create performance issues with large datasets. Furthermore, applications requiring the administration of data that is structured or semi-structured may not be a suitable match for the approach known as relational.

E. Object-Oriented Data Model

Data is provided as objects in the object-oriented data paradigm, and these can be made up of data and processes that operate on it. This method may be used to convey complex data structures and linkages that would be challenging to represent in a relational database. The object-oriented data model is also extremely versatile and can absorb data model changes fast.
The object-oriented data structure, on the other hand, has significant drawbacks. One challenge is that searching in-formation contained in an object-oriented database can be challenging due to the complexity of the links among things. In addition, the object-oriented data model is less well-known than the relational data model, therefore there are not as many resources and tools available for interfacing with object-oriented databases.
To overcome these restrictions, other data models have arisen, such as the NoSQL models. NoSQL models employ non-tabular data structures, which can be more adaptable and scalable than the relational model, including key-value pairs or document-oriented data. The NoSQL models, however, could be trickier to utilize and call for specialized skills or equipment.

TABLE I
Advantages and Disadvantages of Data Models

Data Model	Description	Advantages	Disadvantages
Hierarchical	Tree-like structure of data with parent records and child records	Simple and straightforward; ideal for data with strict parent-child connections.	Capability to manage data with multiple parent-child links or non-hierarchical relationships is

			limited; changing layout after creation is tough.
Network	A hierarchical model extension that allows for more complicated relationships between records; it use a set structure to represent connections.	Supports complex record linkages; suitable for data with a high number of relationships between parents and children.	Reliance on a set structure limits its flexibility; it may be difficult to question.
Relational	Utilizes tables to describe data; most popular data model in database systems	SQL query language is simple and helpful; it facilitates data representation and administration.	Complex entity connections may be difficult to describe; joins and subqueries may cause performance concerns with big datasets; not ideal for unstructured or semi-structured data.
Object-Oriented	Data delivered in the form of objects containing data and processes that act on it; may transmit complicated structures of data and links	Extremely adaptable; can rapidly integrate data model changes	Looking for information may be difficult; the paradigm is less well-known than the relational model; and there are fewer sources and instruments accessible.
NoSQL	Employs non-tabular data structures, such as key-value pairs or document-oriented data	The relational approach is more versatile and scalable.	Utilization is difficult; specialized skills or equipment are needed.

F. Limitations of Data Models

In database systems, various data models are utilized, each with its own set of advantages and disadvantages. When interacting with data with many relationships, the hierarchical data model is excellent for describing data with tight parent-child connections. Although the network data model is an adaptation of the hierarchical data model which makes it possible for more complicated interactions, it may still be difficult to maintain when the data model grows in size. The relational data model is the most commonly utilized data structure in database systems today because it is very versatile and efficient for accessing and processing huge datasets. The object-oriented data paradigm is beneficial for describing com- complicated data structures and connections, however querying data in an object-oriented database may be difficult.

VI. JUSTIFICATION AND HIGHLIGHTS OF CHOSEN MODEL

In light of its extensive usage and usefulness in modern database systems, the Relational Data Model was selected as the major subject of this study. It offers a systematic and organized method for presenting data, making it appropriate for a broad variety of applications such as e-commerce, healthcare, banking, and scientific research. The relational model's easy accessibility and effectiveness in data administration, as well as considerable research and industry use, making it an appealing alternative for investigation.

The relational data model is very structured and best suited for scenarios with well-defined data connections. It is extensively utilized in a variety of sectors and is well-known for its dependability. The relational architecture is an excellent option if your data contains organized information and you need to comply with ACID (Atomicity, Consistency, Isolation, and Durability). We want to give important information on the relational model's significance in handling and analyzing data by focusing on it, but simultaneously recognizing the presence and importance of alternative data models such as hierarchical, network, object-oriented, and NoSQL.

VII. WHAT IS SEMANTICS AND IT'S USES?

A. Semantics

Data's meaning is referred to as semantics. Semantics is used in database systems to define the connections between data pieces. In database systems, there are three different kinds of semantics: syntactic, semantic, and pragmatic. While semantic semantics deals with the meaning of data, syntactic semantics refers to the grammar or structure of data. The application of data in a particular context is the focus of practical semantics. It is difficult to apply semantics to databases because of the complexity of the data and the requirement for precise interpretation.

TABLE II
Semantics Advantages and Disadvantages ([2],[7],[29],[30])

Advantages	**Disadvantages**
Increases the precision and accuracy of data interpretation.	Implementation takes a large amount of time and money.
Allows for the integration of data from several sources.	The complicated nature of semantic technology may be an impediment to entrance.
Improves data consistency and quality	Dependence on standardized terminology and ontologies
Increases data interoperability and sharing	May require significant changes to existing systems
Enables more efficient data retrieval and analysis	Lack of widespread adoption and standardization
Supports advanced applications such as machine learning	Potential for errors in semantic mapping and inference
Provides a common understanding of data across stakeholders	Privacy and security concerns around sensitive data

B. Uses of Semantics

Semantics is widely used in practice across a variety of industries, including healthcare, finance, and retail. In healthcare, semantics is used to manage patient records, allowing healthcare providers to share information and collaborate successfully. In finance, semantics is used to analyze financial data, detect patterns, and make informed choices. In retail, semantics is used to analyze customer behavior and preferences, allowing retailers to offer specific suggestions and improve customer experience.

The semantic web is a World Wide Web expansion that makes it possible to exchange and reuse data among apps, businesses, and communities. RDF, OWL, and SPARQL are just a few of the standards and technologies on which it is built. Numerous advantages of the semantic web include enhanced data interoperability, more automation, and enhanced decision-making capabilities.

C. Importance of Semantics

Semantics is important in database systems because it ensures that data is correctly understood and processed. This is especially crucial in applications requiring data quality and consistency, such as e-commerce, healthcare, banking, and scientific research. Semantics is essential in data modeling, which is the process of developing a model of data objects, properties, and connections. A properly constructed data model helps to guarantee that data is appropriately represented by providing a clear grasp of data semantics.

D. Challenges in Semantics

Dealing with missing or inconsistent data is one of the difficulties in implementing semantics in database systems. Inaccurate query results can have major repercussions in applications like healthcare or finance. Insufficient or unreliable information can also cause queries to fail. It is difficult to apply semantics to databases because of the complexity of the data and the requirement for precise interpretation. The integration of data is one of the main difficulties. It is challenging to combine and analyze data since it is frequently stored in several formats and structures. Data quality is another difficulty. Data is frequently redundant, inconsistent, and incomplete, making it challenging to analyze it correctly. Finally, one major obstacle to using semantics with databases is data security. There must be measures in place to secure data from unauthorized access and to handle privacy issues.

E. Future Directions in Semantics

Semantic data cleansing techniques have been suggested as solutions to this problem. These techniques rely on machine learning to automatically find and fix data mistakes. The caliber of the training data and the precision of the machine learning algorithms determine how effective these techniques are. To address the semantics difficulties, future research should concentrate on establishing improved and more successful data models and query languages. New data models, which include graph-based models, are starting to appear as a viable answer for correctly representing complicated data structures. Graph-based models offer an additional natural method of representing connections between data items and may be used to rapidly query complicated data structures.

The growth of semantic web technology is another topic of future study. Semantic web technologies allow for the combination of data from many sources, making data access and manipulation simpler. They also give a standard method of describing data semantics, which may aid in the reduction of discrepancies in data interpretation.

VIII. WHAT IS QUERY LANGUAGE AND IT'S USES?

A. Query Language

Dealing with sophisticated queries or actions presents another difficulty when attempting to add semantics to database systems. Multiple joins or aggregations may be used in complex queries, making it challenging to optimize for performance. Numerous query optimization strategies have been suggested as solutions to this problem, including the implementation of deep learning algorithms to forecast the best execution strategy for complicated questions.

B. Types of Query Language

Data in a database may be retrieved and changed using a query language. There are multiple query languages, such as XQuery, NoSQL, and SQL. The most used query language is SQL, and it includes indexing and normalization among other optimization methods. NoSQL query languages are built for scalability as well as efficiency in non-relational databases. A query language for XML data, XQuery is used in programs like content management systems and online services.

C. Limitation of Query Language

Using query languages to handle complicated queries with several joins and aggregations is one of the issues. The writing and performance improvement of these kinds of queries might be challenging. Additionally, inconsistent data representation or querying might result from using several query languages for various databases. Several techniques for query optimisation and translation across different query languages have been put forth to overcome these issues. For instance, some systems automatically translate between SQL and NoSQL query languages, and other systems employ deep learning to forecast the best execution strategy for complicated queries.

TABLE III
Advantages and Benefits of Query Language

Advantages and Benefits	Description
Efficient data organization	Data models are theoretical representations of data that enable effective data organization, classification, and administration.
Easier data access and retrieval	Users may use query languages to define the data they want and the requirements for data extraction, enabling data access and retrieval simpler and more effective.
Improved data interpretation and relevance	Semantics enables the establishment of rules and constraints for keeping, retrieving, and modification, as well as the verification of query results' integrity and relevance..

Greater understanding of data	Through data modeling, organizations can gain a deeper understanding of their data and how it relates to their business processes, enabling better decision-making.
Enhanced data security	Data models and semantics can help ensure that data is stored and accessed securely, with appropriate access controls and data validation rules.
Better scalability	Query languages and data models can enable databases to scale efficiently, with the ability to handle increasingly large and complex data sets.
Improved data integration	By standardizing data models and query languages, organizations can enable easier data integration between different systems and applications.

D. Execution of Query Language

The procedure that takes place when a database management system (DBMS) gets data in response to a user query is known as query execution. This process's efficiency is crucial for applications that demand quick and precise access to massive volumes of data. Indexing, caching, and parallel processing are among the strategies offered by researchers to increase query execution speed.

Another option for improving query execution speed is parallel processing. The practice of splitting a big question into smaller sub-queries that may be run concurrently on several processors is known as parallel processing. For big databases, this strategy may drastically decrease query execution times.

IX. SIGNIFICANCE OF DATA MODELS, SEMANTICS, QUERY LANGUAGES

Despite the significance of data modeling, semantics, and query languages, there are still issues that must be solved in order for them to become even more effective tools for data management and analysis. Developing more efficient query processing techniques is one of the most crucial issues. As databases get larger and more complicated, query processing times might become prohibitively slow. As a result, it is critical to create more efficient query processing methods, such as query plan optimization and parallel processing.

X. LIMITATIONS OF DATA MODELS, SEMANTICS, QUERY LANGUAGES

Despite their use, data modeling, semantics, and query languages are not without restrictions. The lack of standardization in data modeling and query languages is one of the primary drawbacks. Distinct database systems use distinct data models and query languages, which may cause consistency concerns when data is shared across systems. The complexity of semantic annotations, which may be challenging to design and maintain, is another restriction. Furthermore, query processing may be complicated and costly in terms of resources limiting database system scalability.

Data models, semantics, and query languages have transformed how we manage, organize, and access data. However, these innovations are not without restrictions. One disadvantage of data models is the fact that they rely on hypotheses that are not always correct or appropriate. Another drawback is that data models may be complicated and difficult to grasp, which can result in database design mistakes.

Semantics has limits as well since there is no single norm for semantic data identification and layout. This makes it difficult to develop compatible semantic information applications that can operate smoothly across various platforms and organizations. Furthermore, semantic data models need extensive data collection and being processed, which may be complicated and resource-intensive.

Query languages' capacity to obtain information from databases may also be constrained. Complex queries with several tables and connections may be challenging to create and comprehend. Furthermore, query languages may be incapable of handling unorganized data or data stored in databases that are not relational.

Despite these constraints, data models, semantics, and query languages will keep influencing human progress in the future. As these technologies mature, new solutions to overcome these constraints and increase their efficacy are expected to emerge.

TABLE IV
Limitations of Query Language ([2], [4], [10], [11], [13], [17], [19])

Limitations and Drawbacks	**Description**
Lack of standardization	Distinct database systems employ distinct data models and query languages, which causes inconsistency when transferring data across systems.
Complexity	Data models and semantic annotations may be difficult to develop and update, and query processing can be time-consuming.
Reliance on hypotheses	Data modelling is based on assumptions that are not necessarily correct or acceptable, resulting in design flaws.
Lack of standard semantic data identification and layout	This makes it challenging to create semantic information applications that work well across several platforms and organizations.
Resource-intensive	Semantic data models need extensive data gathering and processing, and query processing may be resource-intensive, restricting database system scalability.
Difficulty handling complex queries and unorganized/non-relational data	Query languages may be limited in their ability to handle sophisticated queries with several tables and links, and they may not be appropriate for unorganized or non-relational data.

XI. FUTURE PREDICTIONS OF DATA MODELS, SEMANTICS, QUERY LANGUAGES

The upcoming development of data modeling, semantics, and query languages seems to be bright. Artificial intelligence (AI) and machine learning advancements are anticipated to increase query processing precision as well as effectiveness. Furthermore, the creation of new data models and query languages that have become more standardized and accessible may aid in the resolution of compatibility difficulties. Additionally, incorporating the technology of blockchain into database systems is expected to increase data safety and confidentiality. Data models, semantics, and query languages have had a substantial influence on a wide range of industries, particularly training, medical care, enterprise, and politics [30]. However, these systems have several drawbacks, such as limitations in data models, a lack of common semantic standards, and difficulties in constructing complicated queries. Despite these limits, these advancements are still going to play an important role in influencing the growth of humans, and new solutions to overcome these limitations and increase their efficacy are expected to emerge [30]. Semantic search is one such technique, which blends key phrase and vector-based queries to improve the accuracy and relevancy of search results [31]. Semantic search is set to revolutionize query processing and enhance data retrieval in a variety of applications, thanks to advances in NLP operations, object removal, and ontologies/knowledge graphs [31]. According to a recent analysis, data-savvy organizations have the potential for innovation and triumph, and remaining ahead of other companies in 2023 necessitates capitalizing on the most current trends and projections in data models, semantics, and query languages [32].

XII. EFFECT ON HUMAN DEVELOPMENT OF DATA MODELS, SEMANTICS, QUERY LANGUAGES

Human growth has benefited greatly from advances in data modeling, semantics, and query languages. These technologies have transformed several sectors, increasing production, efficiency, and precision. Database systems have helped healthcare practitioners to enhance patient care and increase medical research in the healthcare business. Database systems have aided financial analysts in making educated investment choices in the finance business. Database systems have helped scientists to analyze data and create new discoveries in scientific study. Furthermore, these technologies have offered new employment possibilities in data administration and analysis, which has contributed to economic development.

These advancements in technology have made substantial contributions to the fields of healthcare, education, enterprise, and politics. Data models and query languages, for example, are widely used by healthcare professionals to manage patient data and evaluate medical results. These technologies have allowed the creation of online learning systems that personalize material depending on how students do and their preferences in education.

Data models, semantics, and query languages are critical components of contemporary systems of information, and they have had a significant influence on human progress [28]. These innovations have altered education, healthcare, industry, and government, allowing us to make better judgments and solve complicated issues more quickly. Data models enable us to organize and arrange data in a manner that humans can comprehend and use. Semantics assists us in providing significance to data and making it more intelligible [29]. Query languages are essential for getting data from databases as well as making complicated searches more comprehensible. As these technologies advance, they are expected to play an increasingly greater role in molding human evolution in the future.

XIII. COMPARISON OF DATA MODELS AND QUERY LANGUAGES FOR ELECTRONIC HEALTH RECORDS

Electronic Health Records (EHRs) include a massive quantity of patient data that must be effectively and properly stored, maintained, and accessed. Data models and query languages are critical components of EHR systems because they define the structure, linkages, and access to patient data. However, there is no agreement on the best data model and query language for EHRs. As a result, the purpose of this research is to evaluate and contrast various data models and query languages implemented in EHR systems in order to determine their strengths, limits, and applicability for various healthcare applications.

The worldwide EHR market was estimated at USD 29.7 billion in 2020, according to a report released by the Journal of Medical Systems, and is predicted to increase at a CAGR (compound annual growth rate) of 10.8 percent from 2021 to 2028. This expansion reflects the growing global usage of EHR systems and the demand for efficient data management services [33].

Some EHRs established in 1971 and 1992 were built upon or integrated with hospital billing and appointment scheduling systems, while others, such as COSTAR, PROMIS, TMR, and HELP, were designed as clinical platforms in order to enhance health services and for application in medical research [33].

The majority of EHRs are currently web/client-server-based, rely on relational databases, and pass through data access and input interfaces via mouse-like swiping and cursor devices. The 2009 Healthcare Reform Act boosted the use of electronic health records in the United States. Health data is increasingly being shared electronically from one institution to another, and many medical organizations have developed "EHRs" and "health information exchange (HIE)" networks.

TABLE V
EHR ([33], [34], [35])

Data Model/Query Language	Advantages	Disadvantages	Uses in Healthcare Applications	Year
Relational Model	- Commonly used - Simple to learn and use structured query language (SQL) - High data integrity	- Restricted scalability - Redundant data - Inflexible structure	- Patient management - Clinical decision support - Research - Population health management	2015
Object-Oriented Model	- Highly adaptable than the relational model - Allows complicated data structures - Ideal for complex interactions	- More complicated than the relational model Inadequate stan dardization and interoperability - Prolonged learning process	- Medical imaging - Patient management - Clinical decision support	2016

UCP JOURNAL OF ENGINEERING AND TECHNOLOGY

NoSQL Model	- Extremely flexible - Accepts unstructured information - Performs better with huge datasets	- There is a lack of standardization and compatibility. - Less developed than relational databases - Limited operation and connection support	- Patient monitoring - Clinical decision support - Population health management	2018

Throughout 2015 to 2019, there was a continuing demand for efficient and reliable patient data storage, administration, and accessibility in Electronic Health Records (EHRs). The significance of data models and query languages in establishing the structure, linkages, and accessibility to patient data in EHR systems continued critical. There is not a single" excellent" data format, semantic model, or query language for EHRs since it is dependent on the objectives and demands of the medical facility using the system. Each data model, semantic, and query language has advantages and disadvantages, and choosing the best one needs careful evaluation of the company's objectives and targets, data structure, and method of operation. Relational data models, object-oriented data models, and document-based data models are some of the most often utilized data models in EHR systems. "Structured Query Language" (SQL), "Clinical Query Language (CQL)", and "Fast Healthcare Interoperability Resources (FHIR) Query Language" are all common query languages for EHRs. In terms of semantics, established medical terminologies like "SNOMED CT", "LOINC", and "RxNorm" are critical for correctly describing and obtaining clinical information.

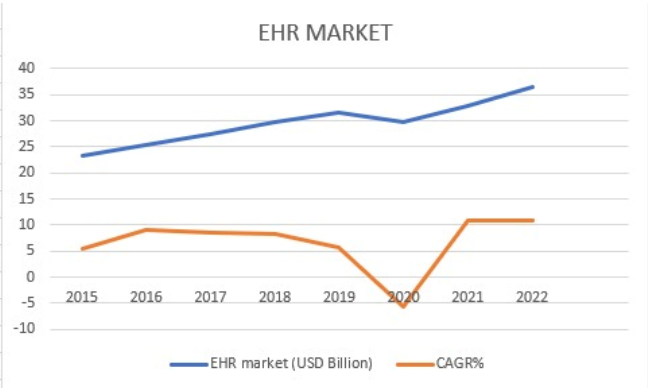

Fig. 1. EHR Market ([37], [38])

Figure 6 depicts the e-prescription ratio in the United States from 2017 to 2021.

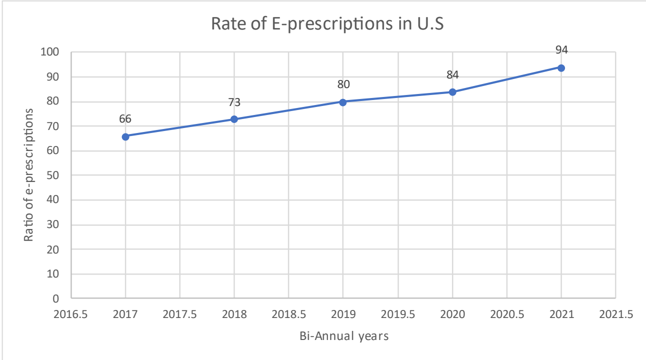

Fig. 2. Rate of e-prescription in US (Statista [36])

With the passage of time, electronic health records and e-prescriptions have improved to the point that they will be used by 94 percent of people by 2021. Whereas just 6 percent of the total go to the physicians in person, the rest go online. This demonstrates the advancement and use of data models, query language, and semantics in the current world.

XIV. CONCLUSION

Modern database systems rely heavily on data modeling, semantics, and query languages. The adoption of proper data models provides data integrity, consistent behavior, and versatility, which are crucial for a wide range of applications including electronic commerce, financial services, healthcare, and research in science. Semantics aids in understanding the significance of data and how it may be utilized to make decisions, while query languages allow for the extraction of relevant information from a database. However, there are still significant obstacles in data modeling, such as creating an adaptable and expandable data model capable of handling complicated data interactions. Future research should concentrate on creating more sophisticated data modeling approaches that may handle these problems while also supporting the increased demand for big data operations. However, there are significant issues in data modelling that need be addressed in future studies. Existing data models, for example, have restrictions that make it challenging to describe complicated data structures. In addition, query languages might not have functionality essential for certain applications. As a consequence, further research is required to address these difficulties and improve the efficiency and effectiveness of data modeling, semantics, and query languages in database systems.

Finally, data modeling, semantics, and query languages have become essential parts of today's database systems, which have altered many industries. However, challenges such as developing greater effective query processing methods and improving the credibility of semantic annotations remain. Future research should focus on solving these challenges to make these tools even more effective for managing and analysing data in a wide range of sectors, including online commerce, health care, financial services, and research in science.

In conclusion, there exists no standard approach for the optimum EHR data model, semantic model, or query language. The decision is determined by the healthcare facility's unique demands and goals. Relational, object-oriented, and record-based data models are some of the most often used data models in EHRs. "SQL", "CQL", and "FHIR Query Language" are common query languages, but established medical terminologies such as "SNOMED CT", "LOINC", and "RxNorm" are essential for correctly defining and collecting clinical information. As the worldwide EHR industry expands, the significance of effective and dependable patient data storage, administration, and accessibility grows.

REFERENCES

[1] Elmasri, R., & Navathe, S. B. Fundamentals of database systems. Pearson, 2019. https://asolanki.co.in/wp-content/uploads/2019/02/Fundamentals_of_Database_Systems_6th_Edition-1.pdf

[2] Winskel, G. The formal semantics of programming languages: An introduction. MIT Press, 1993 https://books.google.com.pk/books?hl=en&lr=&id=JzUNn6uUxm0C&oi=fnd&pg=PR15&dq=Winskel,+G.+(1993).+The+formal+semantics+of+programming+languages:+An+introduction.+MIT+press.&ots=1YqllmXx-r&sig=P4heBkbtWt3JhJsO6mXkzud6gJE#v=onepage&q=Winskel%2C%20G.%20(1993).%20The%20formal%20semantics%20of%20programming%20languages%3A%20An%20introduction.%20MIT%20press.&f=false

[3] Abadi, D. J., Marcus, A. R., Madden, S., & Hollenbach, K. Scalable Semantic Web Data Management Using Vertical Partitioning. PVLDB, 6(7), 485-496, 2013. http://www.cs.umd.edu/~abadi/papers/abadirdf.pdf

[4] Date, C. J. An introduction to database systems. Pearson, 2012

[5] Dignum, V. Semantic web and agent technology: Towards a new synthesis. Springer Science & Business Media, 2011.

[6] Hartig, O., & Bizer, C. SPARQL web query language. Encyclopedia of Database Systems, 2802-2806, 2011. https://dl.acm.org/doi/10.1145/1989284.1989312

[7] Bizer, C., Heath, T., & Berners-Lee, T. Linked data-the story so far. International Journal on Semantic Web and Information Systems (IJSWIS), 5(3), 1-22, 2009. http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf

[8] Chaudhuri, S. An overview of query optimization in relational systems. ACM SIGMOD Record, 27(2), 34-43. 1998. https://web.stanford.edu/class/cs345d-01/rl/chaudhuri98.pdf

[9] Berners-Lee, T., Hendler, J., & Lassila, O. The semantic web. Scientific American, 284(5), 34-43, 2001 https://www-sop.inria.fr/acacia/cours/essi2006/Scientific%20American_%20Feature%20Article_%20The%20Semantic%20Web_%20May%202001.pdf

[10] Connolly, T., & Begg, C. Database systems: a practical approach to design, implementation, and management. Pearson, 2014. http://www.cherrycreekeducation.com/bbk/b/Pearson_Database_Systems_A_Practical_Approach_to_Design_Implementation_and_Management_6th_Global_Edition_1292061189.pdf

[11] G. Lawton, "How to navigate the challenges of the data modeling process | TechTarget," Data Management, Feb. 27, 2019. https://www.techtarget.com/searchdatamanagement/feature/How-to-navigate-the-challenges-of-the-data-modeling-process

[12] Elmasri, R., & Navathe, S. B. Fundamentals of database systems. Pearson, 2015. https://amirsmvt.github.io/Database/Static_files/Fundamental_of_Database_Systems.pdf

[13] Date, C. J. An introduction to database systems. Addison-Wesley, 2013. https://www.mbit.edu.in/wp-content/uploads/2020/05/An-Introduction-to-Database-Systems-8e-By-C-J-Date-CodeBlah.com_.pdf

[14] Codd, E. F. A relational model of data for large shared data banks. Communications of the ACM, 13(6), 377-387, 1970. https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf

[15] Navathe, S. B., & Elmasri, R. Fundamentals of database systems. Addison-Wesley, 1992. https://link.springer.com/book/10.1007/978-3-030-54832-2#page=152

[16] Sheth, A. P. Changing focus on interoperability in information systems: from system, syntax, structure to semantics. International Journal of Cooperative Information Systems, 8(04), 371-393, 1999 https://www.researchgate.net/publication/220040889_Changing_Focus_on_Interoperability_in_Information_SystemsFrom_System_Syntax_Structure_to_Semantics

[17] Silberschatz, A., Korth, H. F., & Sudarshan, S. Database system concepts. McGraw-Hill, 2010. https://d1wqtxts1xzle7.cloudfront.net/54208965/Silber6th.pdf?1503416531=&response-content-disposition=inline%3B+filename%3DSilber6th.pdf&Expires=1681380117&Signature=TBzBcjxAYl~bcS6ugu3~E9a708y5yAzHlrrraogplHJ27TPdc~SBme1A5yEQsE~cpeWwrdR4dV7IKv1KWma6GT5A3ofNFkgtz9c~G4~2fstsl-fBUbwlqb7DiW321gCS4GRBWLbnRbVLlvD4fmbd6zW5HCryk74P-j6Mytn3XiYwiTmnSkvBTHi4vJfF6xXcR5v8vXxPauIV9SPlBuufwL0s5d2VHlkegB~ia-QB6L9ctTRc8kL9bVG-9sjUpM6rV1ru-xBHT3lymFGvexV9Hr~zHDHDEX-d5LIPa3oJGg-nTpMAyFXty8SGnP5H4RVej3N-xcwx0v1Xuexoad38mw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

[18] Agrawal, R., & Carey, M. J. Concurrency control performance modeling: alternatives and implications. ACM Transactions on Database Systems (TODS), 18(4), 461-493. 1993 https://people.eecs.berkeley.edu/~brewer/cs262/ConcControl.pdf

[19] Agrawal, R., & Carey, M. J. Concurrency control performance modeling: alternatives and implications. ACM Transactions on Database Systems (TODS), 18(4), 461-493. 1993. https://people.eecs.berkeley.edu/~brewer/cs262/ConcControl.pdf

[20] Chang, C. C., & Tung, A. K. H. Query optimization in object-oriented database systems. ACM Computing Surveys (CSUR), 30(2), 149-170, 1998 https://cs.uwaterloo.ca/~tozsu/publications/odbms/wonkim/kimchap.pdf

[21] Kifer, M., Lausen, G., & Wu, J. Logical foundations of object-oriented and frame-based languages. Journal of the ACM (JACM), 42(4), 741-843. 1995. https://dl.acm.org/doi/10.1145/210332.210335

[22] Li, J., & Ross, K. A. Performance modeling of disk I/O for database applications. ACM Transactions on Database Systems (TODS), 28(1), 93-131. 2003

[23] Stonebraker, M. The case for shared nothing. ACM Transactions on Database Systems (TODS), 11(1), 4-32. 1986. https://dsf.berkeley.edu/papers/hpts85-nothing.pdf

[24] Zaniolo, C. Object-Oriented Database Systems: Concepts and Architectures. ACM Computing Surveys (CSUR), 22(1), 63-94. 1990. http://reports-archive.adm.cs.cmu.edu/anon/itc/CMU-ITC-103.pdf

[25] Gehani, N. Introduction to database systems. 1990.

[26] Wu and B. Jiinpo, "The effects of data models and conceptual models of the structured query language on the

task of query writing by end users," ProQuest, 1991. https://www.proquest.com/openview/088e79ad1bd1d377cef33e6dc7dd317d/1?pq-origsite=gscholar&cbl=18750&diss=y

[27] Rainer, "Implementing direct manipulation query languages using an adequate data model | SpringerLink," SpringerLink, 1990. https://link.springer.com/chapter/10.1007/3-540-52698-6_14#citeas

[28] "Frontiers | Effect of semantic distance on learning structured query language: An empirical study," Frontiers. https://www.frontiersin.org/articles/10.3389/fpsyg.2022.996363/full

[29] "Understanding the Semantic Data Model," Actian. https://www.actian.com/semantic-data-model/

[30] S. Tiwari and M. A. Jabbar, "Semantic Modeling - an overview | ScienceDirect Topics," ScienceDirect.com | Science, health and medical journals, full-text articles and books., 2022. https://www.sciencedirect.com/topics/computer-science/semantic-modeling

[31] "The past, present, and future of semantic search - Algolia Blog | Algolia Blog," Algolia Blog, Nov. 2022. https://www.algolia.com/blog/ai/the-past-present-and-future-of-semantic-search/

[32] "9 Data Analytics Trends, Predictions, and Resolutions - 2023," ThoughtSpot, 2023. https://www.thoughtspot.com/resources/ebook/nine-data-and-analytics-trends-for-2023

[33] R. S. Evans, "Electronic Health Records: Then, Now, and in the Future - PMC," PubMed Central (PMC), May 2016. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5171496/

[34] J. L. F. Alema´n, I. C. Sen˜or, P. A´. O. Lozoya, and A. Toval, "Security and privacy in electronic health records: A systematic literature review - ScienceDirect," ScienceDi-rect.com — Science, health and medical journals, full-text articles and books., 2013.

[35] J. Schrodt, A. Dudchenko, P. Knaup- Gregori, and M. Ganzinger, "Graph-Representation of Patient Data: a Systematic Literature Review- PMC," PubMed Central (PMC), Mar. 2020. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7067737/

[36] C. Stewart, "E-prescription rate U.S. 2021 | Statista," Statista. https://www.statista.com/statistics/864380/share-of-us-e-prescriptions/

[37] R. and M. ltd, "Ambulatory EHR Market - Growth, Trends, COVID-19 Impact, and Forecasts (2021 - 2026)," Research and Markets - Market Research Reports - Welcome. https://www.researchandmarkets.com/reports/4756853/ambulatory-ehr-market-growth-trends-covid-19

[38] "Healthcare Cloud Computing Market Trends, Growth, Drivers & Opportunities | Markets and Markets," Markets and Markets. https://www.marketsandmarkets.com/Market-Reports/cloud-computing-healthcare-market-347.html