American Journal of Technology Advancement Vol.2, No.3 (Mar, 2025),

E-ISSN: 2997-9382

American Journal of Technology Advancement

https://semantjournals.org/index.php/AJTA

| Research Article

Check for updates

Analysis of Various Technologies for Data Interoperability between Software Tools

PhD. Oybek Allamov

Deputy director for scientific works and innovations

Urgench Branch of Tashkent University of Information Technology named after Muhammad al Khwarizmi Urgench, Uzbekistan

Gulnoza Yuldosheva, Boburbek Babajanov, Bekdiyor Abdusharipov, Fakhriddin Qodirov

Master student, Urgench Branch of Tashkent University of Information Technology named after Muhammad al Khwarizmi Urgench, Uzbekistan

Annotation

This article presents a comparative analysis of four technologies: KNIME, GraphQL, SSIS, and PowerCenter, focusing on key aspects such as security, speed, database interaction, data type handling, and integration with big data platforms. Each technology is evaluated for its strengths in data management and processing, as well as its capabilities in handling large-scale data and ensuring secure operations. KNIME excels in data analysis and transformation with effective speed and resource management. GraphQL is optimized for fast, flexible data retrieval through precise queries, making it ideal for reducing data load. SSIS and PowerCenter, designed for ETL processes, offer high performance and scalability for large data flows and big data integration, with robust parallel processing capabilities. In terms of database interaction, all technologies demonstrate strong support for various databases, but differ in their approaches to handling data types and managing complex data structures. This article explores how each technology can be effectively utilized for data-driven operations, providing insights into their respective strengths and potential use cases.

Keywords: KNIME, SSIS, Power Center, Graph QL, ETL, Rest API, integration, data types, database.

Introduction

Nowadays, the expansion of the network of systems and the compatibility of information between them are becoming more and more important. The joint use of information from different sources and their compatibility in different forms and formats is one of the main problems today. This process is carried out through various software tools and technologies. For example, historical traffic and weather data can be combined to create traffic models. By integrating these data, it is possible to predict future traffic flows[1].

The interoperability of information and the ability to use it together is of great importance, especially in smart cities and smart regions (Smart Cities and Smart Regions). Through intelligent systems, a huge amount of data is collected for various infrastructure, transport, ecology and other fields. However, these data are often not mutually exclusive and isolated, so they cannot work directly with each other. This leads to the creation of isolated, incompatible solutions and systems[4].

On the other hand, by connecting different information systems, it becomes possible to use the existing infrastructure more efficiently. This process not only ensures more sustainable use of resources, but also significantly improves the quality of life of citizens. New and innovative services and programs can be created by using open data. This helps to improve the infrastructure of cities and regions, to modernize them, and to achieve economic and social stability.

materials and methods

Currently, various technologies and software tools are widely used to effectively organize the data interoperability process. Through these technologies, it is possible to collect data from different sources in one system, process them and use them. Tools like KNIME, GraphQL, PowerCenter, and SSIS listed above are being used effectively to solve modern data exchange problems.

Knime.

KNIME (Konstanz Information Miner) is an information analysis program with a graphical interface for automating and optimizing the processes of data analysis, data visualization and data use. This program allows you to work with data and get not only consolidated data from them, but also to perform analysis and statistical operations on various data. It is also distinguished from KNIME by the opportunities it provides in organizing the joint operation of data and performing various analytical processes in them [2].

Reading and writing data: KNIME software has extensive capabilities for reading and writing data. These capabilities include receiving, using, and writing data in many different formats. KNIME allows you to read data in many types of formats, including CSV, Excel, JSON, XML, SQL databases, Google Sheets, PDF files. This makes it possible to work with several sources and combine them. Through KNIME, this data can be converted into the required format and integrated into other programs.

Integration with other systems via REST API: KNIME software has the ability to integrate with other systems via REST API. The REST (Representational State Transfer) API provides access to and exploration of web services. And the KNIME program can receive data, use them, exchange and integrate with other systems through REST ful services. In order to send the results obtained in KNIME to other systems, you can transfer information to other services via the REST API. [5] This enables automated processes to be performed and data to be received.

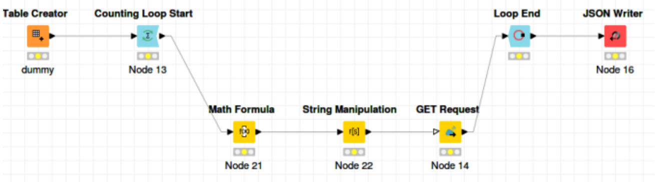

Fig 1. An example of the layout of notes in KNIME.

Figure 2 illustrates the system of all Knime nodes needed to access the Rest API through a single example. In this example, KNIME performs mathematical and analytical operations on the data from the table and performs some actions during the cycle, and writes them to the JSON Writer, which is used to return the API. Data can be integrated into other systems through such methods.

Custom development capability: One of the capabilities of KNIME is the ability to use the KNIME API to expand customization and integration capabilities if there are no existing modules or integrations, and through this KNIME library in code written in programming languages such as Python, Java, or R and there is a possibility of use in similar situations.

These features of the KNIME program provide a wide range of possibilities for data analysis and integration. It allows you to use many different applications and systems without giving you full control over the data. This allows to expand the functionality of the program and better prepare it according to the users' own goals. [6]

GraphQL.

GraphQL is an open source query language developed by Facebook in 2015. It was developed as an alternative to working with the REST API, making data access more flexible and efficient. GraphQL simplifies and optimizes the process of data exchange between the server and the client, especially when working with large volumes of data or complex APIs. [7]

Key features of GraphQL:

- Adaptive Queries: In GraphQL, the client can query the server by specifying only the required data. This allows you to get the necessary information without overloading. Whereas in a REST API, a single endpoint can return entire data, GraphQL returns results that match a specific query.
- Strong Typing: GraphQL allows you to specify a data structure. That is, the user knows in advance which data types and fields are available, and formulates the request accurately. This increases the accuracy between requests and responses.
- Getting information through a single endpoint: While REST APIs have different endpoints for each data source, GraphQL allows you to query all the data you need through a single endpoint. At the same time, dependencies between different resources are also efficiently managed using GraphQL.
- Solves Overfetching and Underfetching problems: In a REST API, the client often faces the problem of getting more or less information. GraphQL, on the other hand, returns only the required amount of data per query. This saves resources and optimizes network usage[3]

GraphQL allows clients to specify exactly what they need, which saves resources and helps to use the network efficiently. It also makes it possible to get all the necessary data through one endpoint, which solves the problem of REST API requiring a separate endpoint for each data. Through Subscriptions, GraphQL allows tracking data updates in real-time, ensuring clients receive the most up-to-date information, and its operation with this principle makes it compatible with real-time mode.

GraphQL is one of the modern data exchange technologies that efficiently organizes data requests between clients and servers. Its flexibility, precise data query and cross-platform usability make it more efficient than REST API. It is widely used in various industries, including web and mobile applications, e-commerce platforms, and microservice architectures. [8].

PowerCenter

PowerCenter is a data integration platform developed by Informatica and widely used to integrate, manage and process data at the enterprise level. PowerCenter provides effective data flow management by automating the collection, cleaning, transformation and management of data from various sources. The platform is widely used by organizations working with large amounts of data (Big Data) for data collection, transformation and loading (ETL – Extract, Transform, Load) processes.

Main features of PowerCenter:

- Integrating Big Data: PowerCenter provides the ability to collect large amounts of data from various sources (databases, files, cloud services, etc.). This integration process is performed with high accuracy and speed.
- Data cleaning and transformation: With the help of the platform, it is possible to automatically clean data, that is, to filter and process incorrect or unnecessary data. This will improve the quality of the data and ensure that further analysis will be more accurate.
- Data transformation and processing capability: PowerCenter is capable of performing complex transformations on data, such as converting data into different formats, performing calculations, or combining data from different sources.
- Ease of use and automation: PowerCenter works through a graphical interface that allows users to easily create and manage ETL processes without writing code. This is especially handy for non-technical users. [9] [10]

The main advantages of PowerCenter are its extensibility and flexibility. PowerCenter is able to work with large volumes of data and provides high flexibility in integration with various technologies, databases and resources. The platform enables reliable data integration at the enterprise level. Data flows are automated and optimized, providing high reliability when working with large volumes of data. Data transformation and integration can be automated in PowerCenter. Users can schedule processes and schedule them to run automatically, saving time and resources. Also, the possibility of data cleaning and transformation increases the quality of data. Accurate and reliable information is essential for further decision-making and optimization of business processes.

Microsoft SQL integration service (SSIS).

A part of Microsoft SQL Server, it contains data, makes changes to it, and provides the ability to transport it from one place to another. In the process of use, it solves the following problems. Using SSIS, data is imported from one or more sources. This data can be in different formats: Excel, CSV, JSON, or from additional relational or cross-system sources, such as Oracle or MySQL. Through SSIS, it is possible to perform data aggregation and transformation. It supports taking data from a database, filtering, combining, converting to aggregates, or data manipulation.

SSIS has powerful capabilities for data transformation. For example, it is possible to format data, combine data sources, perform mathematical calculations and other complex operations. This helps to clean the data and prepare it for further analysis processes. It can also be used to automate and plan data processing processes. For example, users can configure data update or download processes to be performed at a specific time every day or every week. It is very convenient in saving time and resources. Incremental Loading and Change Data Capture (CDC): Applying additional loading techniques and CDC mechanisms can optimize data retrieval and processing in SSIS packages. Technologies such as detecting modified data using CDC, performing additional data extraction using timestamps or additional keys, and scheduling packet execution during off-peak hours can help improve performance. [11][12]

Microsoft SQL Server Integration Services (SSIS) implements an integration service in Microsoft SQL Server to manage ETL (extract, transform, load) processes for reading, writing, and transforming data. SSIS is commonly used to move data, transform, load data from a database, and load data into a database. This system makes it possible to create programs of corporate international level that automate the processes of studying, analyzing and transferring data for different sources and purposes. SSIS is managed through SSMS (SQL Server Management Studio) or Visual Studio and serves to create Microsoft SQL Server packages that execute SQL Server. These SSIS packages are widely used to create ETL processes.

Discussion

Working with the REST API.

KNIME is very easy to integrate with REST APIs. Through its "KNIME REST" module, it is possible to receive and transfer data from various data sources. This module provides a convenient interface and provides the ability to connect to REST APIs using automated workflows. [13]

GraphQL offers a more modern approach to the traditional REST API. It provides an opportunity to obtain data accurately and in the required amount, which is especially advantageous when working with large amounts of data. Unlike the REST API, in GraphQL all requests are made through a single API point.

SSIS is a bit more complex when dealing with REST APIs. Integrating the REST API in SSIS requires customizing to write additional scripts or bind to data sources. However, SSIS has many other strengths but limitations for REST API integration.

PowerCenter has custom modules to integrate with the REST API. With its Data Integration module, data can be retrieved and transmitted via REST API. But these integrations often require additional setup and programming.

While each technology supports a REST API, the process varies in ease of use, flexibility, and request management capabilities. For example, GraphQL is used as a full API query system, while other technologies use REST APIs to retrieve or send data. KNIME, SSIS, and PowerCenter handle GET/POST requests via the classic REST API. While GraphQL is similar to the REST API, it has a flexible query structure that allows you to retrieve only the data you need. All four technologies primarily receive or transmit data in JSON or XML format. Both KNIME and PowerCenter have options for working with JSON or XML. In SSIS, data is processed through a C# or Python script. GraphQL is extremely flexible and allows for multiple data retrievals in a single request, which doesn't overload data, unlike REST APIs. KNIME, SSIS, and PowerCenter, on the other hand, often use more specific and static queries. Ease of input/processing In KNIME, working with the Nodes API is easy, requiring no coding. SSIS uses a REST API using a Script Task, which requires programming experience. PowerCenter, on the other hand, handles API requests using a REST V2 Connection, which is relatively simple but less flexible.

Table 1. Comparison of REST API features across technologies.

Technology	KNIME	GraphQL	SSIS	PowerCenter
REST API support	via the REST Reader node	API request system	Via Script Task	Using REST V2 Connection
Method of obtaining information	GET/POST requests	Adaptive queries	GET/POST requests	GET/POST requests
Data format	JSON, XML	JSON	JSON, XML	JSON, XML
Flexibility	Medium	High	Medium	Medium

Speed.

In KNIME, speed can depend on large amounts of data. For the most part, it processes medium-sized loads quickly and efficiently, but sometimes slows down with very large loads.

GraphQL is very fast, especially if the required data is well defined. But for very large and complex queries, GraphQL can be slow, especially when fetching large amounts of data from multiple sources can slow down the system.

SSIS is a fast and efficient tool for data integration. However, as the amount of data increases, its efficiency may slow down, especially when processing large amounts of data.

PowerCenter can process large amounts of data quickly and efficiently. This tool is well optimized for working with large amounts of data and has high performance indicators.

GraphQL is a technology optimized for high-speed data retrieval that retrieves only the data you need through flexible queries. KNIME, SSIS, and PowerCenter are efficient for working with large amounts of data, but they are used to handle full ETL (Extract, Transform, Load) processes, which can sometimes be relatively slow compared to GraphQL.

Parallel Processing and Data Streams KNIME, SSIS, and PowerCenter have parallel processing capabilities, allowing for faster processing of large amounts of data simultaneously. GraphQL has less parallel processing capability, but efficiency is increased by its response processing tailored to its queries.

Speed in ETL processes SSIS and PowerCenter run ETL processes at high speed, especially efficient when loading and processing large amounts of data simultaneously. KNIME, on the other hand, provides good speed in analysis processes, but its data loading and transformation speed may be slower compared to ETL-oriented technologies. [15] [16]

Table 2. Features of working with datatypes by technology

Technology	KNIME	GraphQL	SSIS	PowerCenter
Data acquisition speed	Good at data analysis	Very fast when the request and response are optimized	Good at handling large amounts of data	Efficient in handling large data streams
Parallel processing	There is parallel processing	Parallel processing is limited	There are parallel ETL processes	Extensive parallel processing is possible
Speed in ETL processes	Good at transforming large amounts of data	Not intended for ETL processes	High speed in ETL processes	Very high speed ETL processes

Working with different datatypes.

KNIME supports many datatypes, including numeric, text, date, and boolean values. Modules for managing datatypes are especially useful for handling large amounts of data easily.

GraphQL puts data into a rigid structure and clearly describes the various datatypes. It supports numbers, text, lists and many other datatypes. Hard-binding these datatypes makes GraphQL queries easy and precise.

SSIS is powerful in working with datatypes. It fully supports a variety of datatypes, including XML, JSON, and numeric data, and processes them through appropriate rules. However, customizing datatypes can sometimes require additional configuration.

PowerCenter is more flexible in working with datatypes. It includes a wide variety of datatypes and automatically handles large amounts of data matching and formatting. This makes it very convenient to work with large amounts of data.

Each technology supports different datatypes. KNIME, SSIS, and PowerCenter support a wide variety of data types, including int, float, double, string, and database-related data types (such as SQL datatypes). GraphQL provides several basic data types (int, float, string, boolean), but datatypes are more defined there and users can create their own datatypes.

In KNIME, datatypes are easily managed through nodes and compatibility between multiple datatypes is done. In SSIS and PowerCenter, datatypes are used in various transformations, and they are mainly associated with the datatypes used in the database. GraphQL is stricter when working with datatypes, and it is required to specify specific datatypes in the request and response format.

Converting and converting datatypes KNIME and PowerCenter are convenient when converting datatypes and automatically perform the conversion between datatypes when managing the flow of data. SSIS uses special conversion functions to convert datatypes. GraphQL doesn't provide the ability to change datatypes, but custom queries are used to work with datatypes customized by the server.

Table 3. Features of working with datatypes by technology

Technology	KNIME	GraphQL	SSIS	PowerCenter
Supported datatypes	Wide range (int, float, string, SQL datatypes, etc.)	Basic datatypes (int, float, string, boolean)	Database related datatypes, int, float, string, etc	Wide range (database datatypes and other common datatypes)
Managing datatypes	Easily controlled by nodes	Defined precisely by the user	Datatypes are used in transform processes	Datatypes are managed during transformation and mapp
Changing datatypes	Automatic adaptation	Datatypes are less likely to change	Through conversion functions	Automatic and manual switching

Safety.

Data encryption using HTTPS (TLS) is provided when receiving data via the KNIME REST API. KNIME provides security by supporting various authentication methods, including OAuth2 and API keys. KNIME Server has authentication and permission settings for users, which means it's possible to define exactly who works with what data. It's also easy to manage security settings when connecting to external APIs in KNIME. If the database or API requires security protocols, they can be configured. [17]

In GraphQL, the extent to which users access data is precisely controlled. Supports authentication tools such as OAuth2 and JWT (JSON Web Token). [18] GraphQL uses rate limiting to prevent users from fetching the required data through multiple queries.

SSIS provides secure connections to databases through Windows authentication or SQL Server authentication. SSIS packages encrypt database connections and SSIS packages. Packages can only be opened and modified by authorized users.

PowerCenter supports strong authentication systems such as LDAP, Kerberos, and Active Directory. Roles and permissions can be clearly defined for each user. PowerCenter has the ability to encrypt data, and HTTPS/TLS protocols are used when data is transmitted. PowerCenter keeps a detailed log of each user's actions, which enhances security and monitoring. Data flow and user actions are monitored through real-time monitoring.

The table below describes a general comparison of the security aspects of these technologies:

Table 4. Comparing technologies security systems.

Technology	KNIME	GraphQL	SSIS	Power Center
Authentication	OAuth2, API Keys	JWT, OAuth2	Windows Auth, SQL Auth	LDAP, Kerberos, Active Directory
Encrypt data	via HTTPS (TLS)	via HTTPS	Connections and packets are encrypted	via HTTPS
Permissions	Based on users and groups	Control access to fields at the user level	Packages and permissions are clearly defined	Permissions and roles are clearly defined

Ability to work with databases.

KNIME provides very comprehensive database integration capabilities. There are many nodes for working with different SQL and NoSQL databases. KNIME supports many types of data, including relational and non-relational databases.GraphQL is not a database per se, but it is often used as a tool for efficient database manipulation. Typically, GraphQL acts as a middle layer to connect to a database and send a request to the DBMS to retrieve or store data.

SSIS has full integration with SQL Server databases. However, SSIS can work not only with Microsoft SQL Server, but also with other databases, including DBMSs such as Oracle, MySQL, and Postgres.

PowerCenter has very powerful database capabilities. It has integration with a wide range of relational and non-relational databases. It is very well configured for data flow management and handles large amounts of data with ease.

All four technologies allow working with different databases, but they differ in the ease of data connection, transformation, and data processing. It can connect directly to KNIME, SSIS, and PowerCenter databases and provides integration through built-in connectors. GraphQL, on the other hand, provides intermediate servers and data connectivity options when working with a database.

Data connection and management KNIME, SSIS, and PowerCenter allow direct connection to multiple databases (SQL Server, Oracle, MySQL, PostgreSQL, etc.). Data acquisition and processing in KNIME is done through a continuous process. In SSIS, a high level of integration is provided in the implementation of ETL (Extract, Transform, Load) processes. GraphQL, on the other hand, allows you to connect to data through an API rather than a direct connection, which makes it more flexible, but requires more middle-layer work.

Transformation capabilities SSIS and PowerCenter have powerful ETL tools that enable large-scale data collection, transformation, and loading (ETL). KNIME also offers powerful tools for analysis and transformation, but lacks the professional ETL tools of PowerCenter and SSIS. GraphQL, on the other hand, provides flexibility in the process of retrieving data rather than processing it.

Supported Databases KNIME, SSIS, and PowerCenter support multiple databases, including SQL Server, Oracle, MySQL, PostgreSQL, and more. GraphQL can use many types of data sources through an intermediate API layer.

Working with large volumes of data PowerCenter and SSIS have powerful tools for working with large volumes of data due to their use in ETL processes. Although KNIME allows you to work with large volumes of data, it does not have special tools like PowerCenter and SSIS for managing high-level large enterprise data flows. GraphQL, on the other hand, is highly efficient in handling large amounts of data using data optimization and flexible queries.

Table 5. Features of working with a database on technologies.

Technology	KNIME	GraphQL	SSIS	Power Center
Integration capability	Easy connection and integration	Via an intermediate API	Extensive integration capability	Strong enterprise integration capability
Connection	Supports multiple DBMS	Connect via API tools	Deep integration with SQL Server and more It also supports DBMSs	Provides deep support for multiple DBMSs
Supported databases	SQL Server, Oracle, MySQL, etc	Any API supported (depends on data sources)	SQL Server, Oracle, MySQL, etc	SQL Server, Oracle, MySQL, PostgreSQL, etc

Working with Big Data.

KNIME works efficiently with big data. It integrates with technologies like Hadoop and Spark, making it easy to analyze large amounts of data. [14]

GraphQL offers good flexibility when working with big data because queries can be precisely controlled. However, GraphQL can sometimes slow down when processing large queries because it acts as a middle layer to collect all the data.

SSIS is used for big data, but it is not as powerful as a traditional ETL tool for large-scale data processing. When processing large amounts of data, efficiency may decrease in some cases.

PowerCenter is perfect for working with large amounts of data. It is well-tuned for the analysis and transmission of large volumes of data, and offers the possibility of parallel processing and automation of data flows.

Table 6. Features of working with Big Data on technologies

Technology	KNIME	GraphQL	SSIS	Power Center
Integration with Big Data platforms	Direct integration (Hadoop, Spark)	Does not integrate directly with Big Data platforms, but optimizes Big Data through API	Database related datatypes, int, float, string, etcIntegration with Microsoft Hadoop and other Big Data platforms	Integration with platforms, such as Hadoop, Spark
Data acquisition and processing speed	Quickly analyzes and transforms large amounts of data	Effective at retrieving large amounts of data through	In ETL processes, large amounts of data are transformed	Datatypes are managed during transformation and mapp

Technology	KNIME	GraphQL	SSIS	Power Center
Integration with Big Data platforms	Direct integration (Hadoop, Spark)	Does not integrate directly with Big Data platforms, but optimizes Big Data through API	Database related datatypes, int, float, string, etcIntegration with Microsoft Hadoop and other Big Data platforms	Integration with platforms, such as Hadoop, Spark
		adaptive queries	quickly	
Parallel processing	The possibility of parallel operation	Parallel operation is less possible	Parallel ETL is available	Extensive parallel processing capability

Conclusion

KNIME, GraphQL, SSIS, and PowerCenter technologies offer various features for data management, big data analysis, and integration. In terms of security, KNIME and PowerCenter provide strong security features for managing large volumes of data, and they meet enterprise-level security requirements. GraphQL is mainly focused on data retrieval and query optimization, and security rules are usually implemented through built-in APIs and authentication methods. SSIS, integrated with Microsoft technologies, also has excellent security capabilities, but handles data through more ETL processes.

In terms of speed, GraphQL is optimized for fast data retrieval and increases speed by retrieving only the required data. KNIME and PowerCenter provide high speed in analyzing and transforming large volumes of data, but their performance varies depending on the size and complexity of the data. SSIS is also capable of performing high-speed ETL processes, processing large amounts of data quickly.

In terms of working with databases, KNIME, SSIS, and PowerCenter are integrated with a wide range of databases and have powerful capabilities in data extraction, analysis, and transformation. GraphQL doesn't work directly with databases, but allows you to get data quickly and in an optimized way through an API.

In terms of working with datatypes, KNIME, SSIS, and PowerCenter support a wide range of datatypes and are efficient at converting them. GraphQL, on the other hand, uses datatypes in well-defined formats and uses custom datatypes when retrieving data through queries.

For working with Big Data, KNIME, SSIS, and PowerCenter have powerful capabilities that provide improved speed and scalability when working with large amounts of data. GraphQL, on the other hand, is not directly integrated into big data platforms, but improves efficiency in obtaining data through optimized queries. In general, each technology has its own special features and strengths: KNIME excels in data analysis, GraphQL excels in fast data retrieval, and SSIS and PowerCenter excel in ETL processes of large volumes of data and integration with big data platforms. ensures efficiency.

References

1. Winter, O. Allamov, B. Boburbek, Q. Fakhriddin, K. Bekmurod, .Allamova, “A Method Based on Intellectual Technologies of Data Interoperability Between Software Tools”, In 2024 IEEE 25th International Conference of Young Professionals in Electron Devices and Materials (EDM) IEEE, pp. 2490–2493, 2024.

2. Fillbrunn, C. Dietz, J. Pfeuffer, R. Rahn, G. A. Landrum, and M.R.Berthold. "KNIME for reproducible cross-domain analysis of life science data." Journal of biotechnology 261, pp. 149-156, 2017.
3. M. Seabra, M. Nazário, G. Pinto, "REST or GraphQL? A performance comparative study", In Proceedings of the XIII Brazilian Symposium on Software Components, Architectures, and Reuse, pp. 123-132, 2019.
4. E.Y. Song, G. J. FitzPatrick, K. B. Lee, E. Griffor, "A methodology for modeling interoperability of smart sensors in smart grids", IEEE transactions on smart grid, 13(1), pp. 555-563, 2021.
5. H. Çelik, A. Çinar, "An application on ensemble learning using KNIME", In 2021 International Conference on Data Analytics for Business and Industry (ICDABI) IEEE, pp. 400-403, 2021.
6. S. Martino, N. Mazzocca, F. R. Torrepadula, L. L. Starace, "Mobility data analytics with KNOT: the KNIME mobility toolkit", In International Symposium on Web and Wireless Geographical Information Systems, pp. 95-104, 2023.
7. P. Roksela, M. Konieczny, S. Zielinski, "Evaluating execution strategies of GraphQL queries", In 2020 43rd International Conference on Telecommunications and Signal Processing (TSP) IEEE., pp. 640-644, 2020.
8. G. Brito, M. T. Valente, "REST vs GraphQL: A controlled experiment", In 2020 IEEE international conference on software architecture (ICSA), pp. 81-91, 2020.
9. Fedotov, A. Akhmetshin,V. Kulakov, "Development of Regulatory Requirements for Calculation of Electrical Loads of Residential Complexes", In 2024 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM) IEEE, pp. 158-162, 2024.
10. Z. Li, S. Liu, J. Zhang, J. Wang, S. Shen, "Research on the Construction of Smart Grid Digital System Based on Big Data Cloud Computing Model Optimization Algorithm", In 2024 IEEE 4th International Conference on Power, Electronics and Computer Applications (ICPECA), pp. 1287-1291, 2024.
11. R. Waeytens, D. Van Hauwermeiren, W. Grymonpré, I. Nopens, T. Beer, "A framework for the in silico assessment of the robustness of an MPC in a CDC line in function of process variability", International Journal of Pharmaceutics, pp. 658,2024.
12. Ayyagiri, A, P.K. Pandian, P. Goel, "Efficient Data Migration Strategies in Sharded Databases", Journal of Quantum Science and Technology, 1(2), pp. 72-87, 2024.
13. S. Rinderle-Ma, J. Mangler, D. Ritter, "Fundamentals of Information Systems Interoperability: Data, Services, and Processes" Springer Nature, 2024.
14. F. H. Taques, C. Chasco, F. H. Taques, "Integrating big data with KNIME as an alternative without programming code: an application to the PATSTAT patent database", Journal of Geographical Systems, pp.1-31, 2024.
15. AlMasaud, S. Sampaio, P. Sampaio, "Mining Data Wrangling Workflows for Design Patterns Discovery and Specification", Information Systems Frontiers, pp.1-24, 2024
16. Khan, S. Jan, W. Khan, M. I. Chughtai, "An Overview of ETL Techniques, Tools, Processes and Evaluations in Data Warehousing", Journal on Big Data, 6, 2024.

17. O. F. Awad, L. R. Hazim, A. A. Jasim, O. Ata, "ENHANCING IIOT SECURITY WITH MACHINE LEARNING AND DEEP LEARNING FOR INTRUSION DETECTION", Malaysian Journal of Computer Science, 37(2), pp.140-154, 2024.
18. S. Amareen, O. S. Dector, A. Dado, A. Bosu, "GraphQL Adoption and Challenges: Community-Driven Insights from StackOverflow Discussions" arXiv preprint arXiv:2408.08363, 2024.