Viewpoint

Data Lake, Data Warehouse, Datamart, and Feature Store: Their Contributions to the Complete Data Reuse Pipeline

Antoine Lamer[1,2,3], PhD; Chloé Saint-Dizier[1,2], MSc; Nicolas Paris[3], MSc; Emmanuel Chazard[1], MD, PhD

[1]Univ. Lille, CHU Lille, ULR 2694-METRICS, Centre d'Etudes et de Recherche en Informatique Médicale, Lille, France
[2]Fédération régionale de recherche en psychiatrie et santé mentale des Hauts-de-France, Saint-André-lez-Lille, France
[3]InterHop, Rennes, France

Corresponding Author:
Antoine Lamer, PhD
Univ. Lille, CHU Lille, ULR 2694-METRICS, Centre d'Etudes et de Recherche en Informatique Médicale
1 place de Verdun
Lille, 59000
France
Phone: 33320626969
Email: antoine.lamer@univ-lille.fr

Abstract

The growing adoption and use of health information technology has generated a wealth of clinical data in electronic format, offering opportunities for data reuse beyond direct patient care. However, as data are distributed across multiple software, it becomes challenging to cross-reference information between sources due to differences in formats, vocabularies, and technologies and the absence of common identifiers among software. To address these challenges, hospitals have adopted data warehouses to consolidate and standardize these data for research. Additionally, as a complement or alternative, data lakes store both source data and metadata in a detailed and unprocessed format, empowering exploration, manipulation, and adaptation of the data to meet specific analytical needs. Subsequently, datamarts are used to further refine data into usable information tailored to specific research questions. However, for efficient analysis, a feature store is essential to pivot and denormalize the data, simplifying queries. In conclusion, while data warehouses are crucial, data lakes, datamarts, and feature stores play essential and complementary roles in facilitating data reuse for research and analysis in health care.

JMIR Med Inform 2024;12:e54590; doi: 10.2196/54590

Keywords: data reuse; data lake; data warehouse; feature extraction; datamart; feature store

Introduction

Over the last few decades, the widespread adoption and use of health information systems (HISs) have transitioned a substantial amount of clinical data from manual to electronic format [1]. HISs collect and deliver data for care, administrative, or billing purposes. In addition to these initial uses, HISs also offer opportunities for data reuse, defined as “non-direct care use of personal health information” [2], such as research, quality of care, activity management, or public health [3]. Hospitals have gradually adopted data warehouses to facilitate data reuse [4,5]. Even if the data warehouse is a popular concept, data reuse is not limited to feeding and querying a data warehouse. In this viewpoint, our objective is to outline the different components of the data reuse pipeline and how they complement and interconnect with each other. This definition is derived from our personal experiences and insights gained through collaboration with colleagues at various institutions [5-8]. Additionally, we draw on the collective experiences shared by professionals in the field, contributing to a comprehensive understanding of data reuse practices in diverse health care settings. The pipeline is illustrated in Figure 1 and detailed below. Table 1 compares characteristics of each component. Last, Multimedia Appendix 1 provides examples of data, structures, and architectures for each component of the data reuse pipeline.

Figure 1. Components of the complete pipeline for data reuse. EHR: electronic health records; ETL: extract-transform-load.

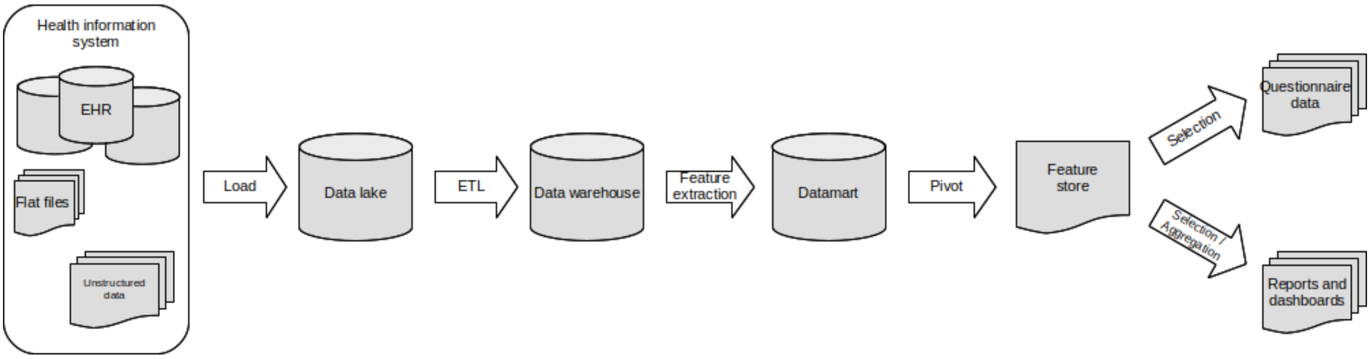

Table 1. Characteristics of each component of the data reuse pipeline.

Characteristics	Software	Data lake	Data warehouse	Datamarts	Feature store
Content	Data and metadata	Data and metadata	Data	Features	Features and metadata about feature
Architecture	Distributed	Centralized	Centralized	Centralized	Centralized
Detail level	Fine-grained	Fine-grained	Fine-grained	Aggregated	Aggregated
Data	Raw	Raw	Cleaned	Cleaned	Cleaned
Nomenclature	Heterogeneous	Heterogeneous	Standardized	Standardized	Standardized
Data model	Normalized	Normalized	Normalized	Normalized	Denormalized
Data structure	Row-oriented	Row-oriented	Row-oriented	Row-oriented	Column-oriented
Purpose	Transactional software purpose	Ad hoc exploratory queries	All purposes	Prespecified purpose	Prespecified purpose

Ethical Considerations

This study does not include human participants research (no human participants experimentation or intervention was conducted) and so does not require institutional review board approval.

Health Information System

The raw data stored in the HIS are distributed across multiple software, making it impossible to cross-reference information between sources due to variations in data formats, ranging from tabular to hierarchical structures and free text [9]. Different technologies and distinct identification schemes for patients, admissions, or any other records compound the complexity. Additionally, direct write access to the software databases is typically unavailable, as software editors rarely grant such privileges to prevent any potential disruption to routine software operation. In transactional software databases, data consist of meticulously organized and highly accurate records presented in rows. These records are collected with great precision to fulfill the specific functions of the software. Alongside the data, a wealth of metadata is also present, including information regarding data collection (eg, information on the individuals inputting data, record timestamps, and biomedical equipment identifiers), as well as software configurations. Notably, a significant portion of these metadata may not be directly relevant to our research purposes, as they primarily support the routine functioning of the software.

The Data Lake

An optional first component of a comprehensive data reuse pipeline is the data lake [10-14]. A data lake is a centralized, flexible, and scalable data storage system that ingests and stores raw data from multiple heterogeneous sources in its original format [12,15]. Data are stored in a fine-grained, row-oriented, and raw format, in a secure and cost-effective environment. These raw data still encompass diverse formats, from structured data to unstructured text documents, images, songs, videos, and sensor data, ensuring that a wide spectrum of information is readily available for various data analytics endeavors [12].

The technologies implemented for the data lake can include the usual relational databases, such as PostgreSQL or Oracle, but also NoSQL databases and big data technologies, such as the Hadoop Distributed File System or Apache Hudi for the storage and Apache Spark, Hadoop MapReduce, or Apache Kudu for the data processing.

Unlike structured data typically integrated into data warehouses, the data lake refrains from immediate structuring or transformation, allowing for a more agile and adaptable approach. This flexibility enables exploration, manipulation, and, if necessary, transformation of the data to fulfill specific research or analytical requirements. By delaying the application of predefined data models, the data lake cultivates an environment where information can be uncovered without predetermined hypotheses. This includes insights that may not have been evident during the initial phases of data collection and storage. The system further facilitates on-the-fly query processing and data analysis [12,15].

In a data pipeline without a data lake, it is essential to finalize the extract-transform-load (ETL) process before leveraging the data. This introduces a time delay, as it necessitates identifying relevant data in the HIS, updating the data warehouse data model for their accommodation, and subsequently designing and implementing the ETL.

In addition, when interpreting the results, if it becomes apparent that relevant data are missing for the analysis, it requires updating both the ETL process and the data model to incorporate the missing data. This iterative cycle of identifying, modifying, and reimplementing can lead to prolonged timelines and may hinder the agility of the data analysis process. Therefore, a data lake approach proves advantageous in providing a more flexible and dynamic environment for data exploration and analysis, potentially avoiding some of these challenges encountered in a traditional pipeline.

The Data Warehouse

The data warehouse stands as the most prevalent component of the pipeline and acts as a centralized repository of integrated data from 1 or more disparate sources [5,8,16-19]. It stores historical and current fine-grained data in a format optimized for further use. This involves a single storage technology, a consistent naming convention for tables and fields, and coherent identifiers across data sources. This is a departure from the data lake where all these elements varied between sources.

The data warehouse is supplied through an ETL process [9,18]. The primary objective of this process is to select and extract relevant data from the HIS or other external resources [19]. During this initial phase, the majority of metadata linked to software operations (such as usage logs or interface settings), monitors, and individuals inputting data are usually excluded. Indeed, these types of metadata do not relate to patient care information and would introduce an unnecessary volume of data. Subsequently, the ETL process enhances the raw data by identifying and correcting any abnormal or erroneous information. Following this refinement, data are integrated into a unified data model independent of the source software [9,19]. Notably, there is a strong focus on harmonizing identifiers from diverse data sources to ensure data integrity and streamline queries involving information from multiple origins. The ETL process is also responsible for regularly updating the data warehouse with new information recorded in the original data sources.

The data warehouse, as a relational database, is typically implemented using systems like PostgreSQL, Oracle, SQL Server, Apache Impala, or Netezza. However, for a data warehouse, exploring NoSQL technologies such as MongoDB, Cassandra, or Couchbase can also be interesting, offering advantages in handling unstructured or semistructured data, and providing scalability for large-scale data storage and retrieval [20]. The ETL process can be developed using 2 types of technologies. The first one, with programming languages such as R (R Core Team), Python (Python Software Foundation), or Java (Oracle Corporation), can be used, coupled with a scheduler like Apache Airflow (Apache Software Foundation), to organize the execution of scripts and retrieval of logs and error messages. The second kind of application is graphical user interface software, such as Talend (Talend) or Pentaho (Hitachi Vantara). They do not require programming capacities, because graphical components, corresponding to data management operations, are organized through a drag-and-drop interface.

To foster collaboration among institutions and facilitate the sharing of tools, methods, and results, several initiatives have emerged to offer common data models (CDM). As a result, table and field names are standardized following a common nomenclature, and local vocabularies and terminologies are mapped to a shared vocabulary. Among these CDMs, the Observational Medical Outcomes Partnership CDM was developed by the Observational Health Data Sciences and Informatics community, which brings together multiple countries and thousands of users [21] and led to methodological and practical advancements [22,23].

As a result, the data warehouse functions as a unified, centralized, and normalized repository, for both fined-grained historical data and metadata, and continues to present information in a row-oriented format. The modeling approach presented by Inmon [24] and described as a “subject-oriented, nonvolatile, integrated, time-variant collection of data” implies that data are stored persistently without any assumptions as to their future use, thus remaining open-ended in their usage.

The Datamarts

While the data warehouse serves as a unique standardized repository, primarily dedicated to data storage, querying these data can be time-consuming due to the volume and distribution of data in the relational model. Furthermore, raw data integrated into the data warehouse may not be readily aligned with specific research or analytical questions, as these data lack the necessary aggregated features. For instance, the data warehouse retains all biological measurements (eg, potassium and sodium), while what will be stored in the datamart are the features related to the biology values, such as the occurrence of hypokalemia, hyperkalemia, hyponatremia, or hypernatremia. Thus, the datamart acts as a dedicated resource for transforming the data into usable and meaningful information [19,25,26]. This transformation process involves feature extraction, achieved through the application of algorithms and domain-specific rules [6,7]. The outcome is data that are tailored to address specific research questions or analytical needs. For instance, within a clinical setting, the datamart can convert raw mean arterial pressure values into a format suitable for detecting perioperative hypotension [5].

Moreover, datamarts can be organized in the form of online analytical processing (OLAP) cubes, offering a multidimensional view of the data [27]. This cubical structure allows for in-depth analysis, enabling users to efficiently explore and navigate across various dimensions such as time, geography, or specific categories, gaining profound and contextualized insights. These datamarts are often modeled in either a snowflake or star schema, optimizing their structure for the creation of OLAP cubes. The star schema, with its central fact table surrounded by dimension tables, or the snowflake schema, which further normalizes dimension tables to minimize data redundancy, both serve to facilitate the creation of these OLAP cubes. Such schemas play a pivotal role in enhancing the efficiency of multidimensional data analysis within the OLAP environment, providing a structured framework for faster and more comprehensive insights.

In the context of health care, an example of an OLAP cube could encompass dimensions such as patient (eg, age and gender); time (eg, admission and discharge dates); medical conditions (eg, primary and secondary diagnoses and medical procedures); hospital unit (eg, information on services, departments, and bed types); health care provider (eg, physicians); and outcome (eg, length of stay, treatment outcomes, and medical costs). The cube would include various facts, such as the number of patients, average length of stay, and average treatment costs. This multidimensional structure allows health care professionals to conduct in-depth analyses, explore trends over time, compare costs across different hospital units, and assess the impact of medical interventions on patient outcomes [19,28].

Datamarts, owing to the structured nature of their data, are typically stored on relational databases (eg, PostgreSQL, Oracle, and SQL Server) [25]. In the case of OLAP cubes, this may include Apache Kylin or other proprietary OLAP tools built on relational databases [28,29].

In contrast to Inmon's [24] approach, the Kimball [9] bottom-up approach places datamarts at the core, with their design driven primarily by business requirements. However, by directly developing datamarts, the Kimball approach may overlook some crucial data that were not initially identified as relevant during the business requirements phase.

As a result, the datamart stands as a centralized component for cleaned and aggregated features for dedicated purposes, still stored in row-oriented structure.

The Feature Store

The feature store addresses the limitations of the traditional row-oriented, relational database structure typically used in datamarts. This architecture, which relies on multiple tables, may not fully meet various analytical requirements. For instance, effective statistical analysis often necessitates a single, flat file with column-oriented variables, mandating the transformation of data from a row-based to a column-based format within the feature store. This process streamlines data access, simplifying complex queries into straightforward selections from a single table. Consequently, the feature store emerges as a centralized repository housing well-documented, curated, and access-controlled features. In addition to features extracted from datamarts, which are often calculated by algorithms derived from business rules, the feature store can also receive features generated by machine learning algorithms [30].

The design of the feature store aims to provide data scientists with direct access to these features, eliminating the need for additional data cleaning, aggregation, or pivoting [31]. This specialized role enhances efficiency and promotes the use of high-quality, analysis-ready data, significantly contributing to the effectiveness of data-driven research in the health care organization. Notably, the feature store not only stores the features themselves but also their associated metadata, documenting how they were calculated and used [31]. It ensures the preservation of all feature versions, guaranteeing the reproducibility of analyses.

When derived from business rules, features are stored in relational databases (eg, PostgreSQL, Oracle, and SQL Server) or in a NoSQL data store such as MongoDB to also store metadata. When features originate from machine learning models, they are stored and shared from big data platforms such as Databricks or Hopsworks [30,32].

As the final component of the data reuse pipeline, the feature store plays a pivotal role in various analytical applications within the health care organization. It significantly contributes to the creation of insightful dashboards and automated reports, delivering real-time and historical information. In research, its most crucial contribution lies in generating denormalized flat tables, similar to questionnaire data tailored for statistical analyses.

Conclusions

In this opinion paper, we propose standardized nomenclature and definitions for the components of a data reuse pipeline. Table 2 summarizes the advantages and limitations of each component in this pipeline.

While the data warehouse serves as a necessary initial stage, the integration of datamarts and a feature store enhances its effectiveness. Datamarts compute pertinent information from raw data, while the feature store organizes it into columns, streamlining data set construction. Additionally, the data lake emerges as a valuable resource for storing raw data in a single location, allowing for exploitation without having to wait for the entire pipeline to be developed.

Notably, in a data pipeline without a data lake, the requirement to complete the ETL process before analysis introduces delays. This involves identifying relevant data in the HIS, adapting the data warehouse data model, and implementing the ETL. Additionally, discovering missing data during result interpretation prompts iterative updates to both the ETL process and the data model, potentially prolonging timelines and hindering data analysis agility.

It is important to emphasize that the specific components and their characteristics described here are not rigidly fixed and can vary based on the unique organizational needs and configurations. For instance, the inclusion of a data lake and feature store is often discretionary, influenced by factors such as the scale and intricacy of source data, the quantity of features, the scope of research projects, the team's size, and the imperative for study reproducibility over time.

Table 2. Advantages and disadvantages of the components of the data reuse pipeline.

Component	Advantages	Disadvantages
Data lake	• All data sources on the same server • Independence from source software • On-the-fly query processing and data analysis without the need for the complete development of an extract-transform-load (ETL) process	• Inconsistencies in data formats and structures • Lack of standard schema can make querying complex • Analyses reproducibility
Data warehouse	• Querying data from both administrative and biology systems is facilitated by the unified data model (ie, data from both systems are linked, and the model conventions are consistent) • Relevant data are retained at the finest level of detail (eg, dates, diagnoses, and all biology values), enabling the answering of numerous questions without necessarily identifying them beforehand	• ETL process must be implemented to standardize the data • Multidimensional data model with several statistical units • Fine-grained data is not directly usable and adapted for statistical analysis or decision-making
Datamarts	• Features are ready to be used directly	• Features are still organized with a row-format (ie, 1 feature per row) in several datamarts
Feature store	• Using features directly, without the need for data management tasks such as joining datamarts or pivoting to reorganize features into columns	• Having developed the entire pipeline beforehand

Conflicts of Interest

None declared.

Multimedia Appendix 1

Comparison of data, structures, and architectures of components of the data reuse pipeline.
[DOCX File (Microsoft Word File), 68 KB-Multimedia Appendix 1]

References

1. Adler-Milstein J, DesRoches CM, Kralovec P, et al. Electronic health record adoption in US hospitals: progress continues, but challenges persist. Health Aff. Dec 2015;34(12):2174-2180. [doi: 10.1377/hlthaff.2015.0992]
2. Safran C, Bloomrosen M, Hammond WE, et al. Toward a national framework for the secondary use of health data: an American Medical Informatics Association white paper. J Am Med Inform Assoc. Jan 2007;14(1):1-9. [doi: 10.1197/jamia.M2273] [Medline: 17077452]
3. Safran C. Reuse of clinical data. Yearb Med Inform. Aug 15, 2014;9(1):52-54. [doi: 10.15265/IY-2014-0013] [Medline: 25123722]
4. Wisniewski MF, Kieszkowski P, Zagorski BM, Trick WE, Sommers M, Weinstein RA. Development of a clinical data warehouse for hospital infection control. J Am Med Inform Assoc. Sep 2003;10(5):454-462. [doi: 10.1197/jamia.M1299] [Medline: 12807807]
5. Lamer A, Moussa MD, Marcilly R, Logier R, Vallet B, Tavernier B. Development and usage of an anesthesia data warehouse: lessons learnt from a 10-year project. J Clin Monit Comput. Apr 2023;37(2):461-472. [doi: 10.1007/s10877-022-00898-y] [Medline: 35933465]
6. Chazard E, Ficheur G, Caron A, et al. Secondary use of healthcare structured data: the challenge of domain-knowledge based extraction of features. Stud Health Technol Inform. 2018;255:15-19. [Medline: 30306898]
7. Lamer A, Fruchart M, Paris N, et al. Standardized description of the feature extraction process to transform raw data into meaningful information for enhancing data reuse: consensus study. JMIR Med Inform. Oct 17, 2022;10(10):e38936. [doi: 10.2196/38936] [Medline: 36251369]
8. Doutreligne M, Degremont A, Jachiet PA, Lamer A, Tannier X. Good practices for clinical data warehouse implementation: a case study in France. PLOS Digit Health. Jul 2023;2(7):e0000298. [doi: 10.1371/journal.pdig.0000298] [Medline: 37410797]
9. Kimball R. The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses. John Wiley & Sons; 1998. ISBN: 978-0-471-25547-5
10. Wieder P, Nolte H. Toward data lakes as central building blocks for data management and analysis. Front Big Data. Aug 2022;5:945720. [doi: 10.3389/fdata.2022.945720] [Medline: 36072823]

11. Madera C, Laurent A. The next information architecture evolution: the data lake wave. Presented at: MEDES'16: The 8th International Conference on ManagEment of Digital EcoSystems; Nov 1 to 4, 2016:174-180; Biarritz, France. [doi: 10.1145/3012071.3012077]
12. Sarramia D, Claude A, Ogereau F, Mezhoud J, Mailhot G. CEBA: a data lake for data sharing and environmental monitoring. Sensors (Basel). Apr 2, 2022;22(7):2733. [doi: 10.3390/s22072733] [Medline: 35408347]
13. Che H, Duan Y. On the logical design of a prototypical data lake system for biological resources. Front Bioeng Biotechnol. Sep 2020;8:553904. [doi: 10.3389/fbioe.2020.553904] [Medline: 33117777]
14. HV S, Rao BD, J MK, Rao BD. Design an efficient data driven decision support system to predict flooding by analysing heterogeneous and multiple data sources using data lake. MethodsX. Dec 2023;11:102262. [doi: 10.1016/j.mex.2023.102262] [Medline: 37448950]
15. Hai R, Koutras C, Quix C, Jarke M. Data lakes: a survey of functions and systems. IEEE Trans Knowl Data Eng. Dec 1, 2023;35(12):12571-12590. [doi: 10.1109/TKDE.2023.3270101]
16. Jannot AS, Zapletal E, Avillach P, Mamzer MF, Burgun A, Degoulet P. The Georges Pompidou University hospital clinical data warehouse: a 8-years follow-up experience. Int J Med Inform. Jun 2017;102:21-28. [doi: 10.1016/j.ijmedinf.2017.02.006] [Medline: 28495345]
17. Chen W, Xie F, Mccarthy DP, et al. Research data warehouse: using electronic health records to conduct population-based observational studies. JAMIA Open. Jul 2023;6(2):ad039. [doi: 10.1093/jamiaopen/ooad039] [Medline: 37359950]
18. Fleuren LM, Dam TA, Tonutti M, et al. The Dutch Data Warehouse, a multicenter and full-admission electronic health records database for critically ill COVID-19 patients. Crit Care. Aug 23, 2021;25(1):304. [doi: 10.1186/s13054-021-03733-z] [Medline: 34425864]
19. Agapito G, Zucco C, Cannataro M. COVID-WAREHOUSE: a data warehouse of Italian COVID-19, pollution, and climate data. Int J Environ Res Public Health. Aug 3, 2020;17(15):5596. [doi: 10.3390/ijerph17155596] [Medline: 32756428]
20. McClay W. A Magnetoencephalographic/encephalographic (MEG/EEG) brain-computer interface driver for interactive iOS mobile videogame applications utilizing the Hadoop Ecosystem, MongoDB, and Cassandra NoSQL databases. Diseases. Sep 28, 2018;6(4):89. [doi: 10.3390/diseases6040089] [Medline: 30274210]
21. Blacketer C. The Book of OHDSI. Observational Health Data Sciences and Informatics; 2021. URL: https://ohdsi.github.io/TheBookOfOhdsi/ [Accessed 2024-11-09]
22. Schuemie MJ, Gini R, Coloma PM, et al. Replication of the OMOP experiment in Europe: evaluating methods for risk identification in electronic health record databases. Drug Saf. Oct 2013;36(S1):S159-S169. [doi: 10.1007/s40264-013-0109-8] [Medline: 24166232]
23. Lane JCE, Weaver J, Kostka K, et al. Risk of hydroxychloroquine alone and in combination with azithromycin in the treatment of rheumatoid arthritis: a multinational, retrospective study. Lancet Rheumatol. Nov 2020;2(11):e698-e711. [doi: 10.1016/S2665-9913(20)30276-9] [Medline: 32864627]
24. Inmon WH. Building the Data Warehouse. Wiley; 1992. ISBN: 978-0-471-56960-2
25. Hinchcliff M, Just E, Podlusky S, Varga J, Chang RW, Kibbe WA. Text data extraction for a prospective, research-focused data mart: implementation and validation. BMC Med Inform Decis Mak. Sep 13, 2012;12:106. [doi: 10.1186/1472-6947-12-106] [Medline: 22970696]
26. Kim HS, Kim H, Jeong YJ, et al. Development of clinical data mart of HMG-CoA reductase inhibitor for varied clinical research. Endocrinol Metab (Seoul). Mar 2017;32(1):90-98. [doi: 10.3803/EnM.2017.32.1.90] [Medline: 28256114]
27. Hristovski D, Rogac M, Markota M. Using data warehousing and OLAP in public health care. Proc AMIA Symp. 2000:369-373. [Medline: 11079907]
28. Vik S, Seidel J, Smith C, Marshall DA. Breaking the 80:20 rule in health research using large administrative data sets. Health Informatics J. 2023;29(2):146045822311805. [doi: 10.1177/14604582231180581] [Medline: 37269132]
29. Ranawade SV, Navale S, Dhamal A, Deshpande K, Ghuge C. Online analytical processing on Hadoop using Apache Kylin. Int J Appl Inf Syst. May 5, 2017;12(2):1-5. [doi: 10.5120/ijais2017451682]
30. Armgarth A, Pantzare S, Arven P, et al. A digital nervous system aiming toward personalized IoT healthcare. Sci Rep. Apr 8, 2021;11(1):7757. [doi: 10.1038/s41598-021-87177-z] [Medline: 33833303]
31. Sen S, Woodhouse MR, Portwood JL, Andorf CM. Maize Feature Store: a centralized resource to manage and analyze curated maize multi-omics features for machine learning applications. Database (Oxford). Nov 6, 2023;2023:baad078. [doi: 10.1093/database/baad078] [Medline: 37935586]
32. Rajendran S, Obeid JS, Binol H, et al. Cloud-based federated learning implementation across medical centers. JCO Clin Cancer Inform. Jan 2021;5:1-11. [doi: 10.1200/CCI.20.00060] [Medline: 33411624]

Abbreviations

CDM: common data model
ETL: extract-transform-load
HIS: health information system
OLAP: online analytical processing

Edited by Christian Lovis; peer-reviewed by R Anand, Ines Reinecke, Najia Ahmadi, Olivier Steichen; submitted 27.11.2023; final revised version received 11.03.2024; accepted 05.04.2024; published 17.07.2024

Please cite as:
Lamer A, Saint-Dizier C, Paris N, Chazard E
Data Lake, Data Warehouse, Datamart, and Feature Store: Their Contributions to the Complete Data Reuse Pipeline
JMIR Med Inform 2024;12:e54590
URL: https://medinform.jmir.org/2024/1/e54590
doi: 10.2196/54590