Data Quality for Semantic Interoperable Electronic Health Records

Shivani Batra
Jaypee Institute of Information Technology University, Sector-128, Noida, India
ms.shivani.batra@gmail.com

Shelly Sachdeva
Jaypee Institute of Information Technology University, Sector-128, Noida, India
shelly.sachdeva@jiit.ac.in

ABSTRACT
The current study considers an example of healthcare domain from a BIG DATA perspective to address the issues related to data quality. Healthcare domain frequently demands for timely semantic exchange of data residing at disparate sources. It aids in providing support for remote medical care and reliable decision making. However, an efficient semantic exchange needs to address challenges such as, data misinterpretation, distinct definition and meaning of underlying medical concept and adoption of distinct schemas. The current research aims to provide an application framework that aids in syntactic, structural and semantic interoperability to resolve various issues related to semantic exchange of electronic health records data. It introduces a new generic schema which is capable of capturing any type of data without a need of modifying existing schema. Moreover, proposed schema handles sparse and heterogeneous data efficiently. The generic schema proposed is built on the top of relational database management system (RDBMS) to aid in providing high consistency and availability of data. For having a deep analysis of proposed schema considering timeliness parameter of data quality, experiments have been performed on two flavours of RDBMS namely row oriented (MySQL) and column oriented (MonetDB). Results achieved favours adoption of column oriented RDBMS over row oriented RDBMS under various tasks performed in current research for timely access of data stored in proposed generic schema.

Keywords: Data Interoperability, Data Quality, Electronic Health Records (EHRs), Generic schema, Column oriented RDBMS, Row oriented RDBMS.

1. INTRODUCTION
In healthcare domain, millions and billions of patient records are recorded on a daily basis. This leads to BIG DATA and thus, demands to handle 3V's (volume, velocity and variety) related to BIG DATA. Current study is focused on handling 3V's for semantic interoperable Electronic Health Records (EHRs). Semantic interoperable EHRs constitute medical data related to various patients that possess same meaning to all users using the same set of attributes. Considering large volume of EHRs, a major portion of storage space accounts for null values. So, handling sparseness can greatly reduce storage space requirements. Moreover, EHRs being related to life of patients, demands real time access to information. This requires various optimization techniques which will help in attaining high velocity. Another important aspect is variety since, EHRs constitutes ambiguous heterogeneous data. Apart from dealing with volume, velocity and variety parameters of BIG DATA, healthcare domain often demands for data interoperability. Frequent data access and semantic exchange among distant healthcare organizations aids in improving quality of care. However, this semantic exchange of data needs to address following challenges.

1. *Data misinterpretation:* An entity in medical domain may have different meaning to different organizations. For example, a hospital (APPOLO) may store body temperature data in degree Fahrenheit and other hospital (FORTIS) stores same data in degree Celsius as shown in Figure 1. When data of APPOLO and FORTIS are exchanged, one can misinterpret data regardless of the fact that both data were presenting a correct state of body temperature for some patient. For handling data misinterpretation, all organizations involved in data exchange must follow same data semantics, i.e., there should be semantic interoperability.

Hospital Name	Body Temperature
APPOLO	104
FORTIS	40

Figure 1. Body Temperature recording at two Hospitals

2. *Distinct set of attributes for same medical concept:* Various organizations have their own mechanism for recording data. For example, a medical concept named as 'Blood Pressure' consists of four types of pressures namely 'Systolic', 'Diastolic', 'Mean arterial' and 'Pulse pressure'. 'Systolic' and 'Diastolic' are two types of pressure recorded using medical instrument while 'Mean arterial' and 'Pulse pressure' stores a calculated value using 'Systolic' and 'Diastolic' data. APPOLO might depicts blood pressure condition (Low, Normal and High) of a patient based on 'Systolic' and 'Diastolic' pressure value. While, FORTIS performs same task of predicting blood pressure condition of a patient based on 'Mean arterial' pressure value. This again leads to difficulty in semantic exchange of data. To have same set of attributes there should be a mechanism that provides structural interoperability. Moreover, each attribute should depict the same set of constraints such as, data type, i.e., system must be syntactic interoperable.

3. *Distinct local schemas*: Every organization customizes his schema to capture the data based on local requirements. For example, APPOLO stores value of 'Systolic' and 'Diastolic' pressure and not 'Mean arterial' pressure. Whereas, FORTIS reserves a column in local database table for 'Mean arterial' pressure and not for 'Systolic' and 'Diastolic' pressure. Thus, semantic exchange requires to perform an operation such as, JOIN to construct a maximal schema that can capture each and every detail. However, this approach is not suitable since it require changes in existing database schema.
4. *Timeliness*: In an emergency situation, extracting patient's medical history from local database without any time delay is crucial. Absence of right data at right time can adversely affect treatment given to the patient.

One solution to avoid data misinterpretation and follow same set of attributes for one medical concept is to adopt a standard based EHRs system. To resolve distinct local schema, all organizations need to commit to follow a common standard schema for data storage for smooth semantic exchange. Moreover, schema adopted should be rich enough in terms of capturing all existing and future data requirements without any restructuring of schema. Current research introduces a new generic schema (in Section 3.2) which helps in achieving schema interoperability. Also, proposed generic schema can also be easily expanded to provide data security to enhance data quality. Physical storage approach (row oriented or column oriented) adopted for the proposed generic schema affect the timely access of data. Experimentations has been done to account for this effect.

1.1 Key Contributions

Core aim of current research is to address issues (data misinterpretation, distinct set of attributes for same medical concept and distinct local schema) related to data interoperability. Key highlights of the work done in this paper are as follows:

1. A new generic persistence model.
2. Minimizing storage requirement by eliminating the need of storing null values.
3. Handling heterogeneous data.
4. Maintenance of various quality parameters such as, accuracy and validity, believability, reliability, security, completeness, accessibility, consistency and fitness for use.
5. Experimental evaluation of proposed generic storage on two variants of RDBMS viz. row oriented RDBMS (MySQL) and column oriented RDBMS (MonetDB) under various tasks performed to have an insight of timeliness as a data quality parameter.

This will help in building an EHR system which is capable of dealing large volume with high velocity and can store variety of data.

1.2 Organization of paper

The paper is divided into various sections. Section 2 describes layered approach used by various standard organizations to aid in standardized EHRs. Section 3 ushers the proposed approach and presents the way syntactic, structural and semantic interoperability is achieved. Moreover, it introduces a new generic persistence to handle sparseness and heterogeneous data. Section 4 explains various data quality parameters achieved in current research. Section 5 highlights the experimental setup and results attained considering timeliness parameter of data quality. Section 6 finally concludes the work done.

2. STANDARDIZED EHRs

Electronic Health Records (EHRs) have a complex structure that may include data from about 100-200 parameters [2][16] such as, temperature, blood pressure and body mass index. Individual parameters will have their own contents. Each contains an item, such as, 'data' (e.g., captured for a blood pressure observation). It offers complete knowledge about a clinical context, (i.e., attributes of data), 'state' (context for interpretation of data), and 'protocol' (information regarding gathering of data), as shown in Figure 2 (depicting completeness). Standardized EHRs aid in providing same context among all organization. For example, value of blood pressure recorded in sitting position might vary from the value recorded in standing position. Using standardized EHRs, state description can be captured to provide same context to data while exchanging.

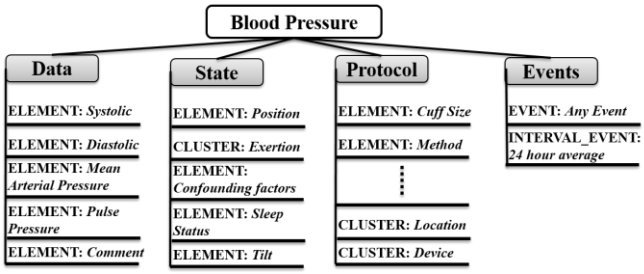

Figure 2. Blood Pressure as a concept (at document level)

Many standard organizations are making constant successful efforts to achieve standardization in healthcare domain for the purpose of semantic interoperability. Some of these famous organizations are Health Level 7 (HL7) [10], European Committee of standardization Technical committee 251 (CEN TC251) [6], International standard organization (ISO) [12] and openEHR [14]. These organizations adopt a layered approach for providing semantic interoperability among EHRs.

2.1 Layered Approach for Standardization

In past, single layer approach was used to design an application. Incorporating single layer hides segregation between programming and domain specific concepts, making an application difficult to modify. Thus, application needs to be rebuild from scratch to accommodate required amendments. For quality enhancement and reduction in repeated efforts for building an updated application, multi-layered approach was introduced.

Multi-layer methodology divides architecture of building an application in multiple layers. Each layer highlights issues related to one of various concepts related to developed application. Considering EHRs, multilevel model approach [1] is widely adopted, aiming for standardization. Standardization is key to achieving fitness for use i.e. communicating same interpretation of data to everyone as originally planned. Layered approach is presented in Figure 3. It divides the application architecture in

different layers termed as Reference Model (RM) [3], Archetype Model (AM) [5] and Service Model (SM) [4].

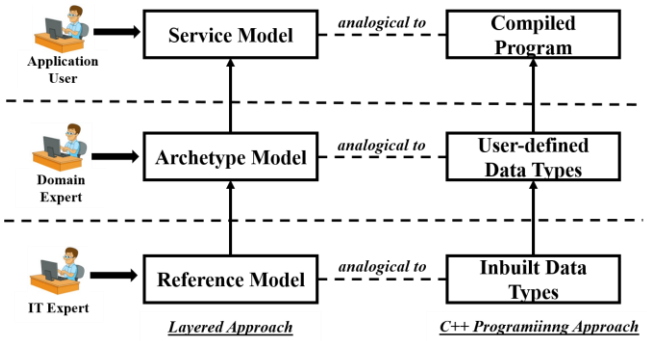

Figure 3. Analogy of layered approach to C++ programming framework

Layered approach can be well-understood through its analogy to the C++ object oriented programming approach. There are inbuilt data types such as, 'int', 'float' and 'char' that are already defined and the application programmer needs to make use of inbuilt data types to define structure or classes as per their needs. RM layer of layered approach depicts inbuilt data types and AM layer depicts user defined data types of C++ object oriented programming approach. SM layer deals with the overall application provided to end user similar to a compiled program in C++ environment.

RM in healthcare domain provides all technical information in terms of data types or data structures to be adopted by final application. This requires involvement of information technology (IT) expert for definitions of various aspects used in RM. RM is stable in nature, similar to inbuilt datatypes in C++ environment.

AM defines domain specific knowledge in form of small modules called archetypes [4]. All knowledge regarding a medical concept resides in an archetype. For example, Blood pressure archetype defines five data attributes termed as 'Systolic', 'Diastolic', 'Mean Arterial', 'Pulse pressure' and 'Comment' with their complete definition (as shown in Figure 2). Various aspects depicting completeness includes data type followed by attribute, domain of attribute, magnitude units in which attribute is defined and links to standard terminologies such as, SNOMED-CT [11] and LONIC [13]. Making use of standard terminologies aids in semantic interoperability by providing a common definition of various terms used in healthcare domain. At this level, domain expert utilizes the information provided in RM to define requirement specific aspects within an archetype. Once an archetype is defined it can be saved in archetype repository and, reused on demand. Healthcare is an expanding domain. As the domain knowledge expands, a new archetype is build using core classes defined in stable RM. Any modification in an archetype or building a new archetype will not require any change at RM or SM. For instance, incorporating any new user defined structure or class in C++ environment will not impact existing inbuilt data types and programs.

SM makes use of archetypes to deliver an application which can further be reused to build any number of user specific applications. For instance, any number of programs can be built in C++ environment using existing inbuilt and user defined data types.

Layered approach was initially proposed by openEHR. Later on other organizations such as, ISO 13606 and HL7 adopted dual model approach for standardization. Interoperability among standards help in providing ability to communicate between various applications built based on various standards. Numerous researches exist [8] that provides framework for switching between different standards. An organization that adopts openEHR, HL7 or ISO 13606 can easily migrate to any standard of choice using such frameworks. Thus, it is very easy for various organisations to adopt same standard (openEHR in our case). Current research focuses on openEHR for achieving data quality and experimentation.

3. SOLUTION APPROACH FOR SEMANTIC EXCHANGE OF EHRs

Semantic exchange of data in healthcare domain is highly demanded to enhance quality of care. The current study aims to resolve the challenges addressed in section 1 for a reliable semantic exchange of data.

1. Data misinterpretation is resolved by linking to standard medical terminologies such as, SNOMED-CT and LONIC. Archetype constraints metric (units) of each attribute that helps in overcoming any misinterpretation. Hence, adopting standard facilitates semantic interoperability.
2. Same set of attributes for same medical concept following same data semantics is achieved through the use of archetypes.
3. Problem of distinct local schema is handled via proposing a new generic schema. Schema adopted capture existing and future data requirements. Simultaneously, it handles sparse and heterogeneous data.

Current research proposes to use application built on archetypes defined by openEHR standard and to persist data in a generic persistence as shown in Figure 4. Each hospital may download any number of archetype from Clinical Knowledge Manager (CKM) [7] based on their local requirements and store them in a local archetype repository. Archetype repository of various hospitals can have distinct set of archetypes, exactly same set of archetypes or overlapping set of archetypes depending upon their local requirements. Based on local archetype repository, a customized clinical application is built for the corresponding hospital.

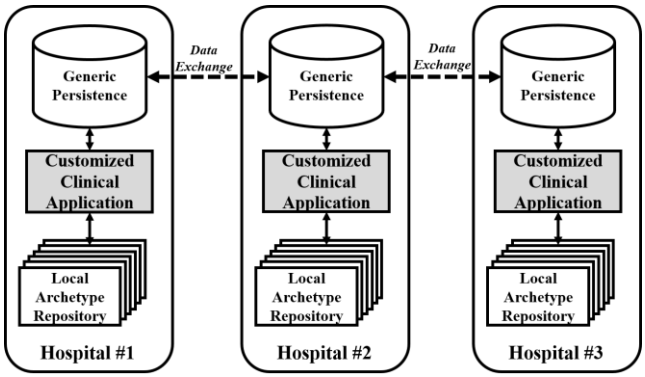

Figure 4. Proposed Approach

Data captured by local application is thus stored in a generic schema proposed in current research (Section 3.2). Generic schema allows migration of data from various organizations without making any changes at schema level. Moreover, proposed generic schema can be enhanced to incorporate security mechanism that in turn helps in enhancing data quality.

3.1 Handling Interoperability

As specified above, different organizations must adhere to same set of attributes for same medical concept. Current research proposes to use archetypes defined by openEHR for this purpose. Archetypes are agreed formal representation of a medical concept. It defines consensus on maximal representation of a medical concept. Archetypes are being defined through Archetype Definition Language (ADL) [5].

For creation and versioning (modifying an existing) of an archetype, a prototype process is followed that involves teams constituting various medical experts. After many review iterations, archetype is agreed to be published in a standard online library such as, CKM. Currently, archetypes available on CKM are followed by 87 countries [7]. Any organization that wishes to make use of an archetype can download it from any online library related to any standard. An archetype downloaded from a standard online library based on one standard can be easily transformed in other standard using tools such as, POSEACLE converter [8] using ontology-based archetype transformation process. POSEACLE converter provides online functionality to transform an ISO 13606 based archetype into an equivalent openEHR based archetype. Providing archetypes in a standard online library enables various organizations involved to easily download archetype anywhere and anytime for their local archetype repository.

As described in Section 2, archetypes are built on a stable structure defined RM. All archetypes related to a standard (openEHR in our case) will follow same RM. This aids in providing syntactic interoperability. Use of archetype provides same set of attributes for same medical concept irrespective of organization adopting it. This provides a mechanism for achieving structural interoperability. Moreover, archetypes are linked with standard terminologies such as, SNOMED-CT and LONIC that aids in providing common data semantics. Achieving common data semantics provides semantic interoperability and resolves issue of data misinterpretation. Thus, archetype provides business rules, ontology, terminology binding, F-logic, versioning mechanism, standardized data definition, content and structure, and language translation capability [17].

Considering all above mentioned features related to semantic exchange, the current research makes use of archetype to propose extension to EAV model.

3.2 Proposed Generic Schema

Various healthcare organizations store data in their own schema as per their local requirements. This creates issues while semantically exchanging data from various independent resources. To overcome this problem, current research proposes to use a generic persistence. Existing solutions for generic persistence in healthcare domains [9] recommend use of Entity-Attribute-Value (EAV) model [15]. EAV constitutes three columns named as Entity, Attribute and Value. One row in EAV corresponds to one attribute value of particular entity (as per relation table). EAV model depicts the same logical representation as relational table through metadata table which reserves information such as, name of all attributes. Use of metadata provides information regarding null values that are not stored in EAV model. Only non-null values are stored in EAV model to limit the storage wastage due to presence of sparse entries. EAV suffers from the issue of heterogeneity due to presence of single value column that constitutes data related to one data type only.

Current research proposes an extension to EAV model namely Archetype Entity Attribute Value (AEAV) to capture archetype based data. AEAV provides a generic persistence for archetype based applications that is more secure than EAV. Firstly, AEAV extends basic three column structure of EAV to four column structure for capturing archetype details also as shown in Figure 5.

Entity	Archetype_Name	Attribute_Name	Value_int
1	Blood Pressure	Systolic	105
1	Blood Pressure	Diastolic	85
2	Body Weight	Weight	70
3	Body Mass Index	Body Mass Index	40
2	Blood Pressure	Systolic	125
2	Blood Pressure	Diastolic	100
4	Body Weight	Weight	65

Entity	Archetype_Name	Attribute_Name	Value_string
2	Blood Pressure	Comment	High
1	Blood Pressure	Comment	Normal

Figure 5. EAV extended to store archetype details

To deal with heterogeneity, AEAV table is divided in multiple tables' segregated based on type of data in value column. Similar to EAV, AEAV also stores only non-null values. After defining the extended EAV storage structure, next step is to define numeric coding for archetype names and attribute names using a manually designed mapping dictionary as shown in Figure 6. Mapping dictionary also serve role of metadata table (as in EAV).

Mapping dictionary defined for AEAV has various advantages as follows:

1. *Improved storage:* Apart from storage enhancements achieved by not storing null values, AEAV optimizes space by not storing long names of archetypes and attributes. It reduces the storage space by replacing the need of storing archetype redundantly and attribute name textually to numeric codes that consumes less space.
2. *Improved searching speed:* Search efficiency in finding codes related to one archetype is high by making use of index structure.
3. *No prior knowledge:* Adding new archetypes in existing system require no prior knowledge about existing codes of attributes since each archetype reuses same set of codes. Combination of archetype name code and attribute name code will define a unique code that serves as a unique identifier in main table.

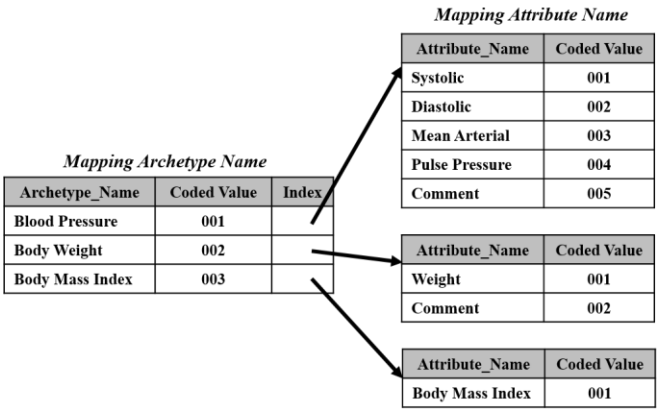

Figure 6. Mapping dictionary for archetype and attribute names

Once mapping dictionary is defined, Archetype_Name and Attribute_Name columns of extended EAV table are replaced by their corresponding codes. Finally, Archetype_Name and Attribute_Name columns are combined as one column named "ArchAtt" using steps as follows.

1. Convert numeric code of Archetype_Name into equivalent 8 bit binary code.
2. Append '00000000', i.e. eight 0 bits to the end of 8 bit Archetype_Name code to make it a 16 bit code.
3. Convert the 16 bit code into an equivalent decimal and replace existing Archetype_Name value with this new value.
4. Add decimal values of Archetype_Name and Attribute_Name columns and replace Archetype_Name and Attribute_Name columns are with one column named ArchAtt containing this summation value.

Final outcome of the above process on initial tables, i.e., AEAV model is shown in Figure 7.

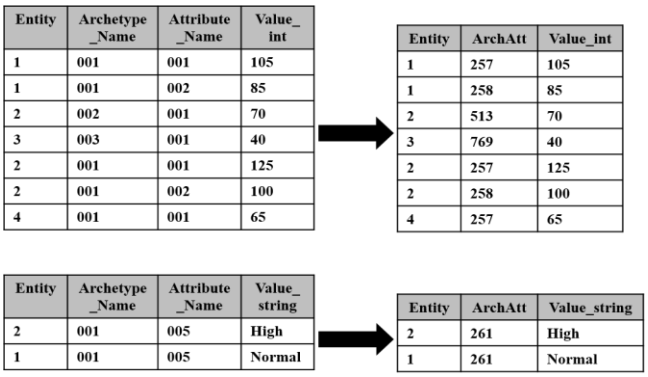

Figure 7. Archetype Entity Attribute Value (AEAV) model

Addition of coded 'ArchAtt' (of AEAV) in place of 'Attribute' column (of EAV) makes AEAV more secure than EAV. AEAV is meaningless until related coding mechanism is known and mapping dictionary are available. Thus, any attack on data will not be able to understand data in absence of mapping dictionary and algorithm followed to combine Archetype_Name and Attribute_Name in ArchAtt.

AEAV can be modified to enhance security feature. To accomplish this, 8 bit code can be replaced with an 'n' bit code in step 2 of coding algorithm. Different organizations can adopt different values of 'n'. This make coded values of one organization distinguished and insignificant to other organizations. Information regarding 'n' can be sent to the organization involved in data exchange through a well-defined secured data encryption mechanism.

Although EAV/AEAV eliminates the need to store null values, it does introduce an overhead of storing entity/attribute code. Thus, there is a trade-off between amount of sparseness and overhead introduced in EAV/AEAV. Larger the amount of sparseness, lower will be the overhead. Hence, EAV/AEAV should be preferred for domain constituting huge volume of null values. Healthcare is one such domain and thus, AEAV is suitable to be adopted for EHRs. Moreover, EHRs are very crucial in terms of ethical and legal issues related to it. This demands for a secure transfer of information. AEAV is step towards the secure transfer of EHRs data.

So far, in current research various challenges identified for semantic exchange of EHRs and solution proposed are summarized in Table 1.

4. DATA QUALITY CONSIDERATIONS

In addition to various parameters (Accuracy and validity, believability, reliability, security, accessibility, completeness, and consistency) of data quality identified in [16], authors in current research commit to achieve fitness for use as a data quality parameter while semantic exchange of data.

1. *Accuracy and Validity:* Use of archetype entitles solution for accurate and valid data. Archetype constitute various business rules to precise data domain and mathematical logics. Applications built on top of archetypes must adhere to these business rules and thus, achieves accuracy and validity.
2. *Believability:* Involvement of domain expert at AM level ensures believability of user in application developed.
3. *Reliability:* Current research adopts dual model approach which segregates responsibilities of an IT expert from a domain expert. This segregation restrict any communication gap between an IT expert and domain expert while designing an application. Moreover, archetypes available on standard online library (such as, CKM) follows a rigorous approach for their development, modification and deployment.
4. *Security:* A generic persistence named AEAV is proposed in current research. Coding of archetype name and attribute name creates a meaningless information in absence of mapping dictionary, number of bits used for coding, and coding algorithm.
5. *Consistency:* Consistency is achieved in current research by adopting RDBMS for persistence of data. RDBMS commits to ACID (Atomicity, Consistency, Isolation and Durability) properties. Any system built on RDBMS, automatically commits to ACID properties.

Table 1. Solution to challenges identified

#	`Challenge	Proposed Solution
1	Data Misinterpretation	• Solved through adoption of archetype based system. • Using archetypes aids in capturing maximum possible information about medical concept. • Archetypes provide links to standard medical terminologies such as, SNOMED-CT and LOINC.
2	Distinct set of attribute for same medical concept	• Archetypes define standard set of attribute for a medical concept. • Archetypes following one standard can be transformed to archetype following another standard using online tools such as, POSEACLE convertor.
3	Distinct Local Schema	• Proposed generic schema, AEAV handles this issue. • Schema is capable to capture all existing and future data requirements without making any changes in schema.
4	Sparseness	• AEAV doesn't store any null value. • AEAV reduces space by eliminating need of storing long archetype and attributes names.

6. *Completeness:* Archetypes are designed to capture maximal consensus on data definition related to a medical concept. This ensure completeness of data. Healthcare domain has one related problem of sparseness, i.e., most of the values are kept null due to many reasons such as, patient don't want to reveal personal information and some parameters are not applicable in certain situations. AEAV is designed to avoid storage of null values and thus, excel in saving storage space. To achieve completeness, mapping dictionary can be used to construct complete set of attributes constituting a medical concept. Knowledge of complete set of attributes facilitates in identifying null values, i.e., the attributes for which no value is found, will be null.
7. *Accessibility:* Current research aims to provide interoperability of data. This enhances accessibility of data.
8. *Fitness of use:* Approach in current research suggests to adopt a standardized EHRs system and generic persistence. Involvement of standardized EHRs restrict to use same set of attribute for same concept. Moreover, generic persistence proposed in current research helps in providing schema interoperability. Thus, data ported from one site to another will depict same meaning to both.

5. EXPERIMENTS AND RESULTS

So far, authors achieved syntactic, structural and semantic interoperability by adopting a standard based system for developing clinical application. Moreover, generic schema proposed in Section 3.2 provides reduced storage requirement (thus, handling more volume), support for variety of (heterogeneous) data and schema interoperability. This section is devoted to show the impact of different physical storage of RDBMS viz. row-oriented and column-oriented on timeliness of data stored as per AEAV. Absence of right information at right time might cause severe losses in terms of patient's health and life. Thus, retrieval of desired data at a very high speed is very crucial in healthcare domain.

5.1 Hardware and Software Used

We implemented AEAV schema using MySQL Workbench 6.0 CE (row-oriented) and MonetDB 5 (column-oriented). All experiments are executed on a pair of 2.66 GHz dual-core Intel Xeon processors with 16 GB RAM running Mac OS X 10.4.11.

5.2 Dataset Description

Data set on which experiments are performed are collected using two different methodologies.

Method #1: Healthcare applications were designed using three archetypes provided by openEHR at CKM, namely openEHR-EHR-OBSERVATION.lab_test-liver_function.v1, openEHR-EHR-OBSERVATION.lab_test-thyroid.v1 and openEHR-EHR-OBSERVATION.blood_pressure.v1 and deployed to three private clinics for collection of data.

Method #2: Liver Disorder dataset and Thyroid dataset were downloaded from the UCI machine learning repository [18-19]. However, it was not standardized. So, user interfaces were developed for related archetypes (openEHR-EHR-OBSERVATION.lab_test-liver_function.v1 and openEHR-EHR-OBSERVATION.lab_test-thyroid.v1). Downloaded data was manually fed in these forms to have dataset as per AEAV schema.

To have persistence as per AEAV in both of the above methodologies, hibernate layer is modified to accommodate coding algorithm of AEAV. Mapping dictionaries were manually populated and provided in the application to achieve coding of archetype name and attribute name at hibernate layer. Attributes present in archetypes but not in UCI datasets were kept null.

In total 75K records were collected as per the relational model. To have reliable and accurate results, we removed redundant records. After removal of redundant data, 75K records were reduced to 50K records.

5.3 Results

Timeliness behavior of AEAV is tested through seven distinguished tasks by formulating queries in MySQL and MonetDB.

Table 2. Impact of row-oriented and column-oriented storage approach on Timeliness of AEAV modelled data

ID	Task	Task Description	Time Taken (seconds)	
			Row-Oriented Storage	Column-Oriented Storage
Q1	Extracting Complete Column Details	Extracting details of Systolic pressure	0.377	0.359
		Extracting Systolic pressure, Diastolic pressure and overall interpretation of all patients	0.618	0.419
		Extracting ALP, AST, ALT, Albumin and Globulins of all patients	5.429	0.577
Q2	Extracting Complete Row Details	Extracting data of all patients	63.684	1.482
		Extracting data of all patients having Total T3 greater than 2	0.499	0.374
		Extracting data of all patients having Systolic pressure greater than 100, Diastolic pressure less than 100 and overall interpretation as Hypotension	0.755	0.569
Q3	Extracting Selected Column Details of Selected Rows	Extracting Systolic pressure, Diastolic pressure and overall interpretation of all patients having Patient ID greater than 4500 and Systolic pressure greater than 100	2.805	1.091
		Extracting ALP, AST, ALT, Albumin and Globulins of all patients having Patient ID less than 5000 and AST greater than 100	4.47	3.962
Q4	Performing Statistical Analysis	Extracting the average value of albumin among people tested for Liver	1.36	0.687
		Extracting number of patients tested for BP and diagnosed with Hypotension	0.396	0.232
		Group the patients tested for Liver according to Albumin values	12.635	0.952
Q5	Adding data	Insert data of one patient	0.774	0.234
Q6	Deleting data	Delete data of one patient	0.372	0.218
Q7	Modifying data	Update data of one patient	0.315	0.297

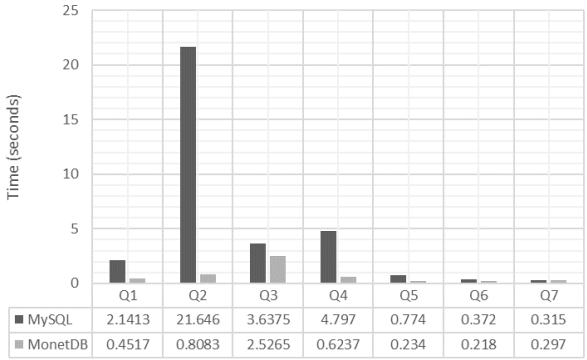

Figure 6. Results of various tasks performed to observe timeliness behavior of AEAV

Seven tasks are extracting complete column details, extracting complete row details, extracting complete row details of selected rows, performing statistical analysis, adding data, deleting data and modifying data. Time taken for performing different tasks is presented in Table 2. Execution time taken by various queries coded for the underlying tasks (presented in Table 2) are averaged and correspondingly presented graphically in Figure 6.

Column details are mostly enquired in EHRs domain to perform analysis of population. The analysis performed might deliver new knowledge that can add value to existing healthcare practices. Storing EHRs data as per AEAV in column oriented physical storage can help in performing faster analysis. Change in physical storage approach of AEAV has shown a drastic performance change in case of extracting rows, i.e., accessing data of specific patients. In summary, under all query scenarios, column-oriented storage approach (MonetDB) outperforms row-oriented storage approach (MySQL).

6. CONCLUSIONS

Considering large scale healthcare data, a clinical information system must be able to handle the 3V's (Volume, Variety and Velocity) of BIG DATA and should simultaneously support data interoperability. Current research propose framework for clinical information system that can handle data misinterpretation, provide same set of attribute for same medical concept in appropriate context and generic schema for persistence. Thus, proposed framework enhances data quality while semantic exchange of data from one organization to another. Moreover, adopting proposed approach will

1. supports syntactic, structural and semantic interoperability,
2. refers a generic schema capable of capturing all current and future data requirements without making any changes in schema,
3. eliminates the need of storing null values to save storage space,
4. supports storage of heterogeneous data,
5. achieves accuracy and validity, believability, reliability, security, completeness, accessibility, consistency and fitness for use as data quality parameters, and

6. improves search efficiency by utilizing optimization techniques of MonetDB.

Current research adopts standard based EHR system to achieve syntactic, structural and semantic interoperability. Despite of attaining syntactic, structural and semantic interoperability, various organizations can adopt their own local schema which is tailored as per their requirements. Thus, efficient data interoperability demands for a generic schema. Hence, a new generic schema is proposed in current research to attain schema interoperability. The generic schema proposed is rich enough to accommodate any existing and future data demands while eliminating the need of storing null values and handling heterogeneous data. Apart from providing schema interoperability, the proposed schema also offers improved security to enhance data quality.

Considering timely access of data, current research observes impact of different physical storage variants of RDBMS, i.e., row-oriented (MySQL) and column-oriented (MonetDB) on proposed generic schema. Experiments are conducted to show the impact on timeliness of data by adopting different variants of RDBMS to store data as per proposed generic schema. Results achieved clearly favors the adoption of column oriented RDBMS over row oriented RDBMS.

7. REFERENCES

[1] Atalag, K. and Bilgen, S. Multi-Level Modeling and the Role of Archetypes in the Design of Health Information Systems: A Modeling Example in Endoscopy. *HIBIT '07 Proceedings of the International Symposium on Health Informatics and Bioinformatics*, May2007.

[2] Atalag, K. and Yang, H.Y. From openEHR domain models to advanced user interfaces: a case study in endoscopy. *In Health Informatics New Zealand Conference*, November 2010.

[3] Beale, T. and Frankel, H. *The openEHR Reference Model. Extract Information Model*. The openEHR release, 1, 2007.

[4] Beale, T. and Heard, S. *openEHR architecture overview*. openEHR Foundation. London, UK, 2008.

[5] Beale, T. and Heard, S. *The openEHR archetype model-archetype definition language ADL 1.4*. openEHR release, 1(2), 2008.

[6] CEN European committee for Standardization: www.cen.eu [online] (Accessed 01/16).

[7] Clinical Knowledge Manager: http://www.openehr.org/ckm/ [online] (Accessed 01/16).

[8] Costa, C.M., Menárguez-Tortosa, M. and Fernández-Breis, J.T. Clinical data interoperability based on archetype transformation. *Journal of biomedical informatics*, 44(5), 2011, 869-880.

[9] Dinu, V. and Nadkarni, P. Guidelines for the effective use of entity–attribute–value modeling for biomedical databases. *International journal of medical informatics*, 76(11), 2007, 769-779.

[10] Health Level 7 International - Homepage: www.hl7.org [online] (Accessed 02/16).

[11] International Health Terminology Standards Development Organization. Systematized Nomenclature of Medicine-Clinical Terms (SNOMED CT). Available from: http://www.ihtsdo.org/snomed-ct/ [online] (Accessed 04/16).

[12] ISO organization: www.iso.org [online] (Accessed 01/16).

[13] Logical Observation Identifiers Names and Codes – LOINC: https://loinc.org/ [online] (Accessed 04/16).

[14] openEHR Foundation: www.openehr.org [online] (Accessed 01/16).

[15] Patrick, J., Ly, R. and Truran, D. Evaluation of a persistent store for openEHR. *HIC 2006 and HINZ 2006: Proceedings*, 2006, 83-89.

[16] Sachdeva, S. and Bhalla, S. Semantic interoperability in standardized electronic health record databases. *Journal of Data and Information Quality (JDIQ)*, 3(1), 2012, 1-37.

[17] Sachdeva, S., Yaginuma, D., Chu, W., & Bhalla, S. Dynamic generation of archetype-based user interfaces for queries on electronic health record databases. *In Databases in Networked Information Systems, Springer Berlin Heidelberg*, 2011, 109-125.

[18] UCI Machine Learning Repository: Liver Disorders Data Set: https://archive.ics.uci.edu/ml/datasets/Liver+Disorders (Accessed 06/15).

[19] UCI Machine Learning Repository: Thyroid Disease Data Set: https://archive.ics.uci.edu/ml/datasets/Thyroid+Disease (Accessed 06/15).