ЕВРО
БАЛКАН

ЗБОРНИК НА ТРУДОВИ

Четврта меѓународна научна конференција
„ Науката - подршка на развојот во Југоисточна Европа “

Скопје 23-24 декември 2016

ЗБОРНИК НА ТРУДОВИ: Четврта меѓународна научна конференција
„ Науката - подршка на развојот во Југоисточна Европа “

Организатор: Институт за дигитална форензика
Универзитет „Евро-Балкан“ - Скопје

Уредник: Проф. д-р Сашо Гелев

Издавач: Универзитет „ЕВРО-БАЛКАН“ Скопје
Република Македонија
www.euba.edu.mk

CIP - Каталогизација во публикација
Национална и универзитетска библиотека "Св. Климент Охридски", Скопје

001.3:330/378(497.7)(062)

МЕЃУНАРОДНА научна конференција (4 ; 2016 ; Скопје)
Науката - поддршка на развојот во Република Македонија : зборник на трудови / Четврта меѓународна научна конференција, Скопје 23-24 декември, 2016 ; [уредник Сашо Гелев]. - Скопје : Универзитет "Евро-Балкан", 2017. - 115 стр. : илустр. ; 30 см

Текст на мак. и англ. јазик. - Фусноти кон текстот. - Библиографија кон трудовите

ISBN 978-608-4714-18-7

а) Научен развиток - Општествени науки - Македонија - Собири
COBISS.MK-ID 105395722

Програмски одбор

- **Проф. Д-р Митко Панов, Универзитет Евро Балкан – Претседател**
- **Проф. Д-р Сашо Гелев – Електротехнички факултет Радовиш Универзитет Гоце Делчев Штип, Република Македонија - Копретседател**
- **Проф. д-р Влатко Чингоски, Електротехнички факултет Радовиш Универзитет Гоце Делчев Штип, Република Македонија**
- **Проф. Д-р Божо Крстајиќ, Електротехнички факултет - Подгорица, Црна Гора**
- **Проф. д-р Илан Садех, Израел**
- **Проф. Гоце Митревски, Аубурн Универзитет, Аубурн, САД**
- **Проф. Ахмед Ајтач, Селџук Универзитет,Конија, Турција**
- **Проф. Кубилај Акман, Ушак Универзитет, Ушак, Турција**
- **Проф. Светлана Антова, Бугарска Акаемија на Науките, СОфија, Бугарија**
- **Проф. д-р Здравко Скакавац, Факултет за правне и пословне студије, Универзитет УССЕ, Нови Сад;**
- **Проф. д-р Лада Садиковиќ, Факултет за криминалистика, криминологија и безбедност, Универзитет во Сараево;**
- **Проф. д-р Гордан Калајџиев, Правен факултет, Универзитет Св. Кирил и Методиј – Скопје, Република Македонија**
- **Проф. Д-р Никола Протрка, Полициска академија, Загреб, Република Хрватска**
- **Проф. Д-р Стефан Сименов, Академија за внатрешни работи на Република Бугарија**
- **Проф. д-р Весна Матијашевиќ Покупец, Универзитет Евро Балкан**
- **Доц. д-р ВангелНоневски, Универзитет Евро Балкан**
- **Доц. д-р Роман Голубовски, Природно математички факултет, Универзитет Св. Кирол и Методиј Скопје, Република Македонија**
- **Доц д-р Костадин Дуковски, Некст Левел Колсалтинг Скопје**
- **Д-р Зоран Нарашанов, Винер осигурување, Скопје, Република Македонија**
- **Проф. д-р Марјан Николовски, Факултет за безбедност, Универзитет Св. Климент Охридски, Битола, Република Македонија**

Организацискиодбор

- **Проф. д-р Сашо Гелев, – Електротехнички факултет Радовиш Универзитет Гоце Делчев Штип, Република Македонија, претседател;**
- **Доц. д-р Мимоза Клекоска, Универзитет Евро Балкан, Република Македонија, член;**
- **Проф. Д-р Божо Крстајиќ, Електротехнички факултет - Подгорица, Црна Гора, член**
- **Доц. д-р Снежана Черепналковска Дуковска, Универзитет Евро Балкан, Република Македонија, член**
- **Проф. д-р Весна Матијашевиќ Покупец, Универзитет Евро Балкан Република Македонија, член**
- **Доц. д-р Вангел Ноневски, Универзитет Евро Балкан, Република Македонија, член**
- **Проф. Гоце Митревски, Аубурн Универзитет, Аубурн, САД, член**
- **Проф. Денис Химчи, Универзитет „Александар Џувани“, Елбасан, Албанија, член**
- **Проф. Ахмед Ајтач, Селџук Универзитет,Конија, Турција, член**
- **Проф. Кубилај Акман, Ушак Универзитет, Ушак, Турција, член**
- **м-р Игор Панев, Универзитет Евро Балкан, Република Македонија, член;**
- **Зорица Каевиќ, Универзитет Евро Балкан, Република Македонија, член**
- **Ивана Гелева Универзитет Евро Балкан, Република Македонија, член**

ПРЕДГОВОР

Позади нас е уште една конференција „Науката-поддршка на развојот во Југоисточна Европа одржана од 23 до 24 декември 2016 година во Скопје, Конференцијата е со наслов Науката – поддршка на развојот во Југоисточна Европа.

Пред четири години за прв пат ја организиравме оваа конференција со цел студентите од вториот и третиот циклус на студии да се оспособат за пишување и презентирање научно-стручни трудови, а останатите учесници да ги пренесат своите најнови истражувања во посочените области.

Програмскиот одбор и реценентскиот тим изврши селекција и овде се презентирани само прифатените трудови. Пред Вас се 13 квалитетни трудови презентирани во 4 секции

За следната конференција ќе се потрудиме да имаме поголем број на трудови и секако трудовите да бидат поквалитетни.

Проф. Д-р Сашо Гелев

.

СОДРЖИНА

Aleksandar Sokolovski,

Neotel – Macedonia, R&D Department, Skopje, Macedonia

Saso Gelev,

Faculty of Electrical engineering, University of Goce Delcev, Stip, Macedonia

Big Data Management practical optimization and implementation of algorithms for the 21 century data evolution (near real time) data processing for the data intensive application.

Abstract: *The scope of this research paper is one very important aspects nowadays, the security and management of one big data, the data in today information bases world play mayor roll in all aspect of business. In this paper, a data evolution model of Virtual DataSpace (VDS) is proposed for managing the big data lifecycle. Firstly, the concept of data evolution cycle is defined, and the lifecycle process of big data management is described. Based on these, the data evolution lifecycle is analyzed from the data relationship, the user requirements, and the operation behavior. Secondly, the classification and key concepts about the data evolution process are described in detail. According to this, the data evolution model is constructed by defining the related concepts and analyzing the data association in VDS, for the capture and tracking of dynamic data in the data evolution cycle. Then we discuss the cost problem about data dissemination and change. Finally, as the application case, the service process of dynamic data in the field of materials science is described and analyzed. We verify the validity of data evolution modeling in VDS by the comparison of traditional database, dataspace, and VDS. It shows that this analysis method is efficient for the data evolution processing, and very suitable for the data-intensive application and the real-time dynamic service.*

Keywords*: Big Data; Lifecycle; Virtual DataSpace (VDS); Data Evolution*

I. INTRODUCTION

Recently, the big data management issues [1, 2] have become increasingly prominent. How to obtain valuable knowledge from the massive and heterogeneous data which distributed in different servers of different regions is the key issue needs to be resolved. For resolving this problem, researchers need to face the challenges of big data management, depth study the related technical program of big data processing, and adopt the scientific and reasonable method to effectively organize the distributed complexity data, so that provide the accuracy and efficient service.

Big data usually has characteristics such as massive, distributed, heterogeneous, association complexity, real-time changes, and so on. Thus the big data management faces the challenges like distributed processing, semantic integration, association mapping, timeliness, and so on. For example, in the field of materials science, the “timber” could be used for two major categories, which are the building materials and the road transport materials. The concept about “timber” is expressed as “roof panels” in building materials, and is expressed as “wooden bridge” in road transport materials. Thereby the related attribute data about “timber” are distributed in two different regions. Therefore, researchers must use some kind of technical approach to construct the semantic associate for the heterogeneous and distributed data. Meanwhile, when some of the source data have changed, it should be able to access the updated data timely. Based on this, the related research about data evolution analysis in the lifecycle of big data would become increasingly important.

In order to more deeply research the modeling approach and dissemination process about data evolution, a concept of Virtual DataSpace (VDS) [3] has been proposed. Virtual DataSpace is the sets of data, services and their relationships, which related with the subject, and based on the supporting of virtualization processing and dynamic evolution. Around the background of massive, distributed and heterogeneous data, which have the features of complex association, rapid growth and real-time changes, the traditional database management mode could not satisfy the needs of data analysis and processing. Compared with the existing data management methods, VDS has the obvious technological advantages in the aspects of model, operations, objects, relations, and construction costs. The VDS model has several unique and significant features, such as the "data first" modeling ideology, the emphasis of data associated mapping and dynamic evolution, the highlights of importance about real-time service, and so on. As a new research field, VDS not only proposes the new modeling method about big data management, but also represents the new idea about dealing with the continually changed data. Therefore, the technical method of VDS is very suitable for researching the data evolution process, which could effectively solve the problem of complex association, timely response the dynamic changing, and finally realize the data reuse and on-demand services by managing the big data lifecycle.

In this paper, the main object is to research the data evolution regularity in VDS for managing the big data lifecycle. Through analyze the data evolution process, construct the evolution model of VDS, which support the achieving of timely capture about the dynamic data in distributed environment. In section 2, the related works about big data and dataspace technical are introduced. In section 3, the definition, feature and process description about big data lifecycle are described. In section 4, mainly analyze the data evolution method based on the idea of big data lifecycle, and construct the data evolution model in VDS. In section 5, the application case about data evolution in material field is described, and verified the validity of data evolution model by comparing with the traditional methods. In section 6, the simple conclusion and further work are given.

II. RELATED WORKS

Recently, related research about big data has just started. Many aspects of theory study and practice work which involved in the new research background should face the new challenges of data management because of the new features shown by the big data technology. S. Wang et al. enumerated several important characteristics [4] that the big data analysis platform required, and summarily described the challenges about big data architecture. A. Cuzzocrea et al. provided an overview about the research issues and achievements in the field of big data analysis [5], and then discussed the problem of multidimensional data analysis about big data. Z.X. Qu [6] proposed a new algorithm based on the auxiliary of semantic graph for the more efficient and intelligent processing of big data. But the semantics of big data are described with RDFs, it has limited the ability of semantic representation. A. Simonet et al. proposed the active data [7] as a programming model, which could alleviate the complexity of data lifecycle and automatically improve the expressiveness of data management applications. However, this model has not considered the data characteristics of semantic association, and has not formed the refined ideology about data evolution.

Researchers are trying to seek a new technology to deal with the new challenges of big data management. The concept of “dataspace” was proposed by M. Franklin et al. in 2005 [8]. Followed by this, researchers have designed wide variety of dataspace architecture and model, and then have proposed several systems which consistent with their respective needs, such as the iMeMex [9], Semex [10], PAYGO [11], OrientSpace [12], UDI [13], etc. These dataspace prototypes are more or less satisfied the demand features of big data processing, such as the semantic integration, pay-as-you-go, on-demand services, schema mapping, etc. However, most of them are limited in the coarse-grained architecture research and the model construction; rarely considered the evolution issues of data model. Based on this, we proposed the concept and method about Virtual DataSpace (VDS) [3]. This technology could make the physical data into virtualization processing, and achieve the dynamic evolution of model by data association modeling in order to realize the query service efficiently. Nevertheless, the evolution mentioned in our previous research mainly means the model’s evolution in VDS, which is arisen by the user feedback. It has not yet been researched for the data evolution, which is generated not only by the user requirement but also by the changing of data source, especially combined with the concepts of big data lifecycle.

About the related concepts of data lifecycle, C. Hedeler et al. proposed a conceptual lifecycle of dataspace [14]. Nevertheless, it only limited to a rough process framework and a comparison between the existing prototypes in the aspects of dimension in dataspace. I. Elsayed et al. proposed a semi-autonomous semantic integrated system in scientific dataspace based on the concept of e-science lifecycle [15, 16], which could analyze the relationships among scientific data sets, and has been used for the scientific work about the breath gas analysis research. The above two system both lack of more in-depth data evolution analysis in detail.

In summary, considering the effect of user requirement and the changes of data source, the data evolution would cause a chain reaction. In the environment which supports the complex association and the distributed storage, the changes of data mode and the quality of data services are essential. Therefore, it is necessary to deeply analyze the data evolution procedure of VDS in detail for managing the big data lifecycle.

III. BIG DATA LIFECYCLE MANAGEMENT

The rapid development of web application demands brought the explosive growth of data. That led to the concept proposing of big data and the research expansion of related technologies. Big data is the frontier technology about data analysis and processing; it could be described as the information assets which has the high growth rates, supports the diversified processing, provides the more optimized technology method, and digs out more valuable knowledge from the massive data. In short, the ability of quickly obtaining the valuable information from the various types of data is the big data technology. Big data has the following features: the volume of data is huge; variety, heterogeneous, distributed, and associated complexity; dynamic incremental change; optimizational processing velocity, i.e. the fast speed and the high efficiency of data processing; get the high value density knowledge from the low value density data.

Big data needs the new processing mode to be able to possess the more powerful decision-making, the more indepth insight, and the more optimized processing capability. Correspond to the features of big data, we proposed the concept of Virtual DataSpace (VDS) [3] to organize, manage and deal with such data. VDS could be described as the sets of data, services and relationships, which related with the subject and based on the supporting of virtualization. VDS has the characteristics such as the “data first”, the associated evolution, the distributed virtualization processing, the service mode of pay-as-you-go which is incremental and ondemand, etc. These technical characteristics are well matched to the features of big data. Therefore, the technology of VDS is very suitable for solving the various issues of big data management.

Considering big data always centers on the idea of “data”, the most important issue of big data management is the research of lifecycle. Big data lifecycle could be defined as the whole evolution process of data and their relationships from generating to vanishing. The evolution process of big data lifecycle could be described as three angles.

A. Data and their Relationships

Figure. 1 illustrates the evolution process of big data lifecycle by considering from the angle of data and their relationships. Firstly, create or produce new data as the beginning of big data lifecycle; then initial allocate the data combined with the semantic description of concept. Secondly, match data from the angle of data attributes or instance, and build the association mapping combined with the relationships between data. Next, integrate the associated data to the global model based on the ontology technology. Followed by it, evaluate the initial construction of data and their relationships. If some stage exist problem, back to the particular stage, and correct the related content; if no problem, then enter the next stage. In the use phase of data, continuously improve the above corresponding stages by according to the situation of actual use, user feedback and data source maintenance. Finally, if exist the possibility of data failure, data obsolescence, or other issues, then judge whether data is invalid. If it is not invalidated, return to amendments; and if it is invalid, make the data demise as the ending of big data lifecycle.

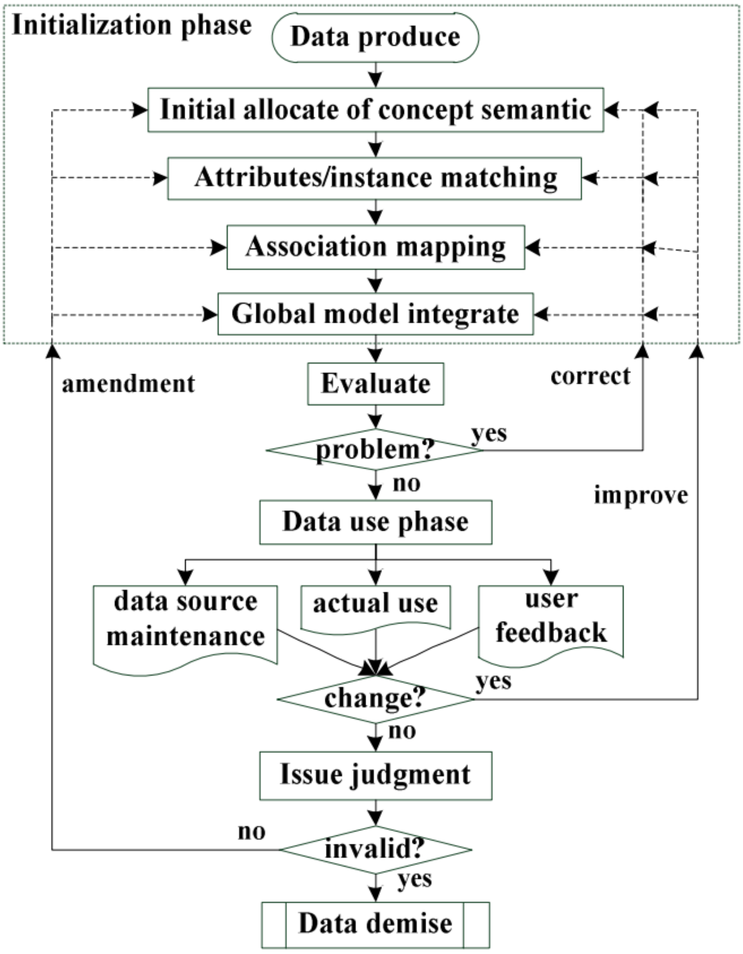


Figure 1. Evolution process of big data lifecycle considering from the angle of data and their relationships.

B. User Requirements and Application Services

Figure. 2 illustrates the evolution process of big data lifecycle by considering the user requirements and application services. Firstly, capture the relevant data and association when discovered the user requirements, then express the changes which are caused by the user demands. Judge the semantic consistency of data which are changed, if inconsistent, retune to re-express the data changes. Otherwise, further proceed to control the data quality, then implement and spread the changes of data. Next, verify whether meet the needs, if not satisfy the user’s needs, return to re-capture the data and association; else further control the version of data service. Finally, support the user services, and find the new requirements based on the new round of application. As such incremental cycle, and gradually improve the big data services.

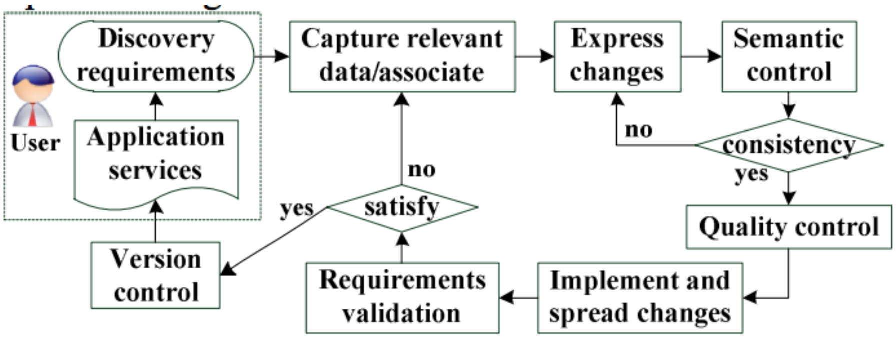


Figure 2. Evolution process of big data lifecycle considering from the angle of user requirements and application services.

C. Operational Behavior

Considering the user operational behavior, the evolution process of big data lifecycle mainly involved the analysis of user operational behavior such as the data addition, the data modification, the data deletion, the information query, the frequency of visits, the last access time, the situation of neighboring access, and so on. These behaviors would influence the trend of evolution process. Therefore these behaviors must be taken into consideration about the research of big data lifecycle. Figure. 3 illustrates the evolution process of big data lifecycle by considering from the angle of operational behavior. Firstly, capture the user operational behavior based on the access log. Then, change the data association according to the specific operation. Simultaneously, the changes of data and their relationships could further cause the changes of data source; and so on ad infinitum.

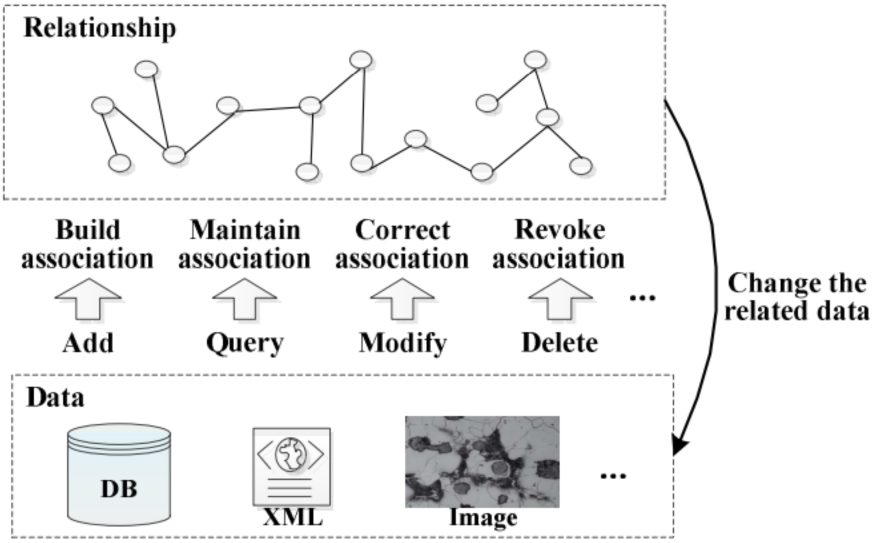
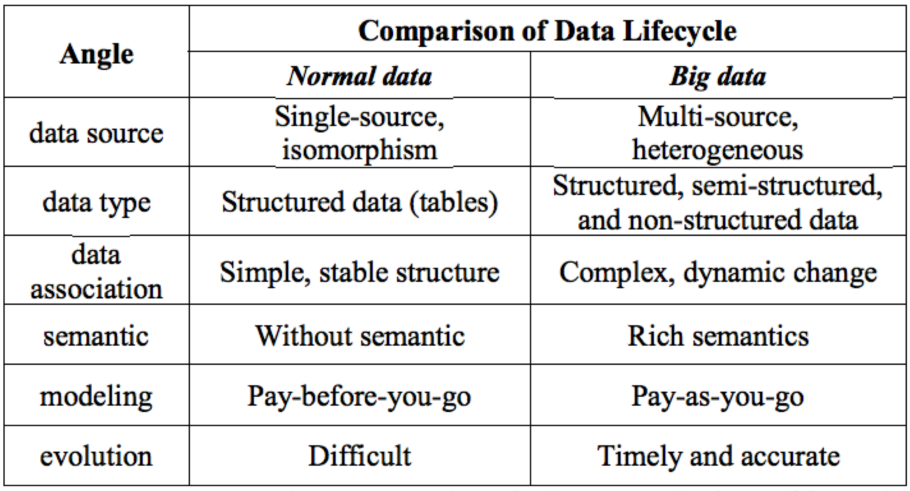


Figure 3. Evolution process of big data lifecycle considering from the angle of operational behavior.

According to the above description about data lifecycle, compare the characteristics of big data with the normal data (Table ĉ). It shows that the big data lifecycle management mainly has the following features.

- Multi-source and heterogeneous data distributed in different regions.
- Not only includes the structured data as tables in the traditional database, but also includes the semistructured data as XML, and the non-structured data as images.
- Complex data association, and change dynamically.
- Rich semantics with data and their relationships.
- Improve the data model incrementally in the whole data lifecycle, therefore has the good extensibility.
- The effect of data evolution is timely and accurate.



Angle	Comparison of Data Lifecycle	
	Normal data	***Big data***
data source	Single-source, isomorphism	Multi-source, heterogeneous
data type	Structured data (tables)	Structured, semi-structured, and non-structured data
data association	Simple, stable structure	Complex, dynamic change
semantic	Without semantic	Rich semantics
modeling	Pay-before-you-go	Pay-as-you-go
evolution	Difficult	Timely and accurate

TABLE I. COMPARISON OF DATA LIFECYCLE BETWEEN BIG DATA AND NORMAL DATA

In summary, the research of data evolution lifecycle could effectively support the managing of big data which is distributed and diversified. The related theories of data lifecycle could effectively direct the data evolution analysis which aims to provide efficient data service. Considering the construction process of VDS could perfectly fit for the lifecycle process of big data, therefore it is necessary to analysis the data evolution process in VDS.

IV. DATA EVOLUTION ANALYSIS IN VIRTUAL DATASPACE

A. Classification of Data Evolution

According to the above description of evolution procedure, and combined with the VDS model features and its dynamically incremental construction demand, the concept of Data Evolution Cycle (DEC) is proposed for managing the big data lifecycle. The DEC in VDS is defined as the whole process of data evolution from produce to demise, which includes the related concept description, analysis methods, logical organization, modeling idea, processing algorithms, etc. during this process. Considering from different perspectives, data evolution could be divided into different types.

From the viewpoint of motivation, data evolution could be classified as three types: the evolution guided by the user requirement feedback; the evolution guided by the actual using behavior; and the evolution guided by the data source maintenance. Their starting points are respectively the user demand, the operational behavior, and the change of data source.

Around the content of evolution data, it mainly includes the changes of domain concept, attribute, instance, value, relationships, and so on. It also could be divided into three types as the original data contents, the derived new data from the evolution process, and the configuration data.

From the viewpoint of evolution manner, data evolution could be classified as two types. One is the manner of manual proposal, which is passive for system, and is the evolution manner of explicit externalization. The other is the manner of automatic discovery, which is proactive for system, and is the evolution manner of implicit internalization.

Considering the evolution mode, data evolution mainly includes the changes such as the addition, deletion, modification, union, merging, mixture, etc. They are the description about the data change patterns. Because of these diversities of change feature, it might generate different effects for the data evolutionary modeling.

B. Key Concepts in Data Evolution Cycle

From the above mentioned process and classification of Data Evolution Cycle (DEC), we can describe the following key concepts which mainly involved in the research domain of data evolution.

Following 1 to 8 are the concepts about data content in DEC

Concept 1: Original data, the data already existed before evolution. It could be the structured data from tables of database, the semi-structured data such as XML, or the nonstructured data such as image, document, video, etc.

Concept 2: Derived data, the data derived from the evolution process. It could be the intermediate data which have the short lifecycle; and also could be the resulting data which have the bran-new meaning that distinguished from other data.

Concept 3: Configuration data, the data existed in the distributed software systems, such as the system configuration, the log parameters, the modeling parameters, and so on. This type of data equally possible would affect the specific data evolution; therefore it should be involved in the unified deployment.

Concept 4: Meta concept, the data which have the abstract connotation. It could be the core concept such as “metal material”, and also could be the specific concept such as “stainless steel”. As long as this concept has its own specific categories, it is the meta concept.

Concept 5: Instance, the data which have the concrete connotation. It could be the specific categories such as “stainless steel” when the corresponding meta concept is “metal material”, and also could be the specific grade such as “12Cr18Ni9” when the corresponding meta concept is “stainless steel”.

Concept 6: Attribute, the data which describe the characteristics of meta concept or instance. For example, “density” could be the attribute about meta concept “stainless steel”, and also could be the attribute about instance “12Cr18Ni9”.

Concept 7: Value, the data which describe the specific attribute value such as numerical value, numerical range, text, etc. For example, “7.9 g/cm3 ” is the value of attribute “density” about instance “12Cr18Ni9”, and “7.6~8.2 g/cm3 ” is the value of attribute “density” about meta concept “stainless steel”.

Concept 8: Relationship, the data which describe the relations between the data contents. They contain the meta concept, instance, attribute, value and the relationship itself.

From the concepts of 4 and 5, it could be seen that some of the meta concept and instance could convert to each other in the different description environment. Fig. 4 illustrates an example of conversion between meta concept and instance. In the relationship tree of meta concept and instance, the root node could only be the meta concept, the leaf nodes could only be the instances, the other nodes could either be the root node, or be the leaf nodes.

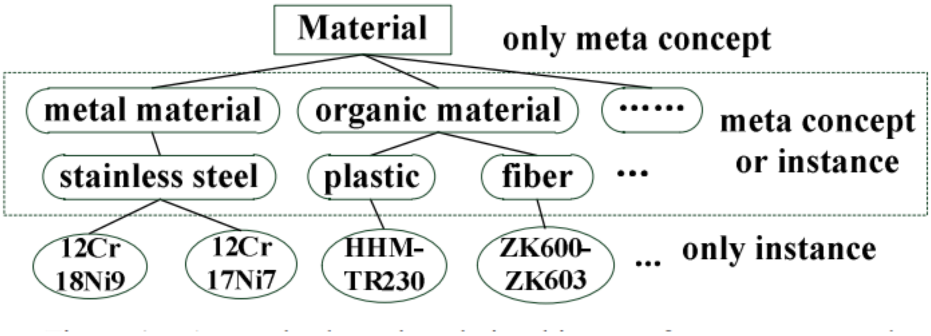


Figure 4. A sample about the relationship tree of meta concept and instance.

Following 9 to 14 are the concepts about evolution mode in DEC, i.e. the data operational behavior.

Concept 9: Addition, the operation of adding data, such as add a new data A.

Concept 10: Deletion, the operation of deleting data, such as delete an existing data B.

Concept 11: Modification, the operation of modifying data, such as modify the data from A to B.

Concept 12: Union, the operation of joining several different data together, such as AB+CD=ABCD.

Concept 13: Merging, the operation of merging several similar data together, such as AB1+AB2=AB.

Concept 14: Mixture, the operation of mixing several overlapping data together, such as ABC+BDE=ABCDE. Following 15 and 16 are the concepts about evolution manner in DEC.

Concept 15: Explicit evolution, the evolution manner of manual proposal externalization, which is passive for system, such as the user feedback, the modification of data source by managers, and so on.

Concept 16: Implicit evolution, the evolution manner of automatic discovery internalization, which is proactive for system, such as the data evolution caused by the behavior analysis of log records. Following 17 to 21 are the concepts about analysis modeling in DEC.

Concept 17: Change, all the data change situations occurred in the modeling process of data evolution, such as the addition, deletion, modification, etc.

Concept 18: Dissemination, all the data dissemination situations occurred in the modeling process of data evolution, such as the influence of changing for the related data.

Concept 19: Trace, all the data evolution tracks in the modeling process. It could be the data change trace by analyzing from the time dimension, and also could be the data dissemination trace by analyzing from the dimension of dataspace.

Concept 20: Semantic divergence, the data semantic inconsistent generated in the modeling process of data evolution. For example, “timber” could be expressed as “roof panels” in the building materials, and also could be expressed as “wooden bridge” in the road transport materials.

Concept 21: Version filtration, carry out the effective control for the confusing problem of data version, which generated in the modeling process of data evolution.

In general, the above key concepts mainly described the Data Evolution Cycle (DEC) from four aspects: data content, evolution mode, evolution manner, and analysis modeling. In which the concepts from 1 to 8 have limited the classification of data content. The method and process of evolution modeling are different for the different types of data content. Thus it is necessary to firstly divide the data content according to the concepts. Concepts 9 to 14 have described some common evolution mode of data in lifecycle. The data could change and move in different directions for different operational behaviors. Considering the different evolution manners as explicit or implicit in concepts 15 and 16, build the specific evolution model by different ways. Concepts 17 to 21 illustrated the core concepts of analysis modeling in DEC. The modeling process of data evolution mainly involves the following problems: the data change with time, the data dissemination with space, the semantic divergence, etc. Therefore, link up the whole data lifecycle from the data content to the evolution method, then to the analysis modeling. Eventually support the construction of data evolution model according to these key concepts.

C. Data Evolution Modeling

According to the above classification and concept description about data evolution cycle in VDS, analyze the data logical organization and build the evolution model by adopting the scientifically reasonable methods. The core modeling ideas of data evolution are the “data first” and the “dynamic association evolution”, which due to the management of VDS, and allocate the data resources on demand by using the semantic mapping and dynamic evolution mechanism. Thus it is needed to find the relationships between different kinds of data by defining the related concepts and analyzing the deeper associations in detail. Thereby construct the data evolution model to support the effective management and optimization treatment about big data lifecycle in VDS.

1) Conceptual model First of all, combined with the above conceptual description about data content, especially concepts 4 to 8, build the conceptual model of data evolution.

Definition 1: Meta Concept is a four-tuples, MC = (mc_name, mc_ontomap, mc_relationship, mc_operation). Where mc_name possess the semantic feature, it is a tuple, mc_name = (mc_name_this, mc_name_ontology). The data semantic consistency could be managed by “mc_name_ontology” as the unified and unique identification. Thereby support the solving of problem about semantic divergence.

The mc_ontomap is a tuple, mc_ontomap = (mc_ontomap_service, mc_ontomap_weight), the mapping between the data evolution model and the user requirement model could be built by “mc_ontomap”, thereby get through the top to bottom contact in VDS.

The mc_relationship is a triples, mc_relationship = (mc_rela_conc, mc_rela_inst, mc_rela_attr), where mc_rela_conc, mc_rela_inst and mc_rela_attr have the similar tuple mode as the definition 5 that described in following.

The mc_operation is a five-tuples, mc_operation = (mc_oper_name, mc_oper_origin, mc_oper_user, mc_oper_time, mc_oper_version), where mc_oper_origin is an N-tuples, mc_oper_origin = (operorigin1, operorigin2, …, operoriginN), it denotes that this meta concept comes from these data. The mc_oper_user could be some person, or automatically generated by the machine. When the operation is “addition”, make the mc_oper_version as 1.

Definition 2: Instance is a four-tuples, Inst = (inst_name, inst_ontomap, inst_relationship, inst_operation). It is very similar to the definition of meta concept, except the inst_relationship is a four-tuples, inst_relationship = (inst_rela_conc, inst_rela_inst, inst_rela_attr, inst_rela_value).

Definition 3: Attribute is a four-tuples, Attr = (attr_name, attr_ontomap, attr_relationship, attr_operation). It is also very similar to the definition of meta concept, except the attr_relationship is a four-tuples, attr_relationship = (attr_rela_conc, attr_rela_inst, attr_rela_attr, attr_rela_value).

Definition 4: Value is a four-tuples, Value = (value_type, value_content, value_relationship, value_operation). Where value_relationship is a triples, value_relationship = (value_rela_inst, value_rela_attr, value_rela_value). The value_type could be numeric, text, image type, document type, etc. The value_content needs to be defined as different tuple modes according to the different value type. For example, if the type of value is numeric, the value_content is a four-tuples, value_content = (numeric_definite, numeric_max, numeric_min, numeric_unit); if the type of value is text, the value_content is just the text content; if the type of value is image, the value_content is a five-tuples, value_content =(image_name, image_type, image_location, image_size, image_description); and so on.

Definition 5: Relationship is a seven-tuples, Rela = (rela_name, rela_type, rela_ontomap, rela_content, rela_weight, rela_relationship, rela_operation). In which, the rela_name, rela_ontomap and rela_operation have the similar tuple mode as the mc_name, mc_ontomap and cm_operation which are described in definition 1. The rela_type could be the father, son, brother, neighbor, similar, opposition, etc. The rela_content has the similar tuple mode as the mc_oper_origin that described in definition 1. The rela_weight denotes the weight value among the relational contents in this relationship. The rela_relationship denotes the relationship between data relationships, it is very similar to the definition of relationship, and i.e. it is also a seventuples as a nested definition about Rela.

2) Evolution method

Figure. 5 illustrates the evolution process of data content about big data lifecycle in VDS. Through analyzing the above concepts and definitions, it could be found that there exists relationships between meta concept and meta concept, meta concept and instance, meta concept and attribute, instance and instance, instance and attribute, instance and value, attribute and attribute, attribute and value, value and value. Particularly, there possible exists the further relationship between one relationship and another relationship. For example, there might exist a strong possibility that the “father” relationship and the “son” relationship possess the “reverse” relationship, only if the both ends of relationship just are corresponding. Around the “rela_type”, “rela_content” and “rela_weight” in definition 5, through the relational construction by analyzing the type, content and weight of relationships, it is able to promote the evolution of data association in VDS. If one data changed, another data which are associated with it would generate the corresponding changes. Thus based on the data association, triggered a series of evolution and dissemination.

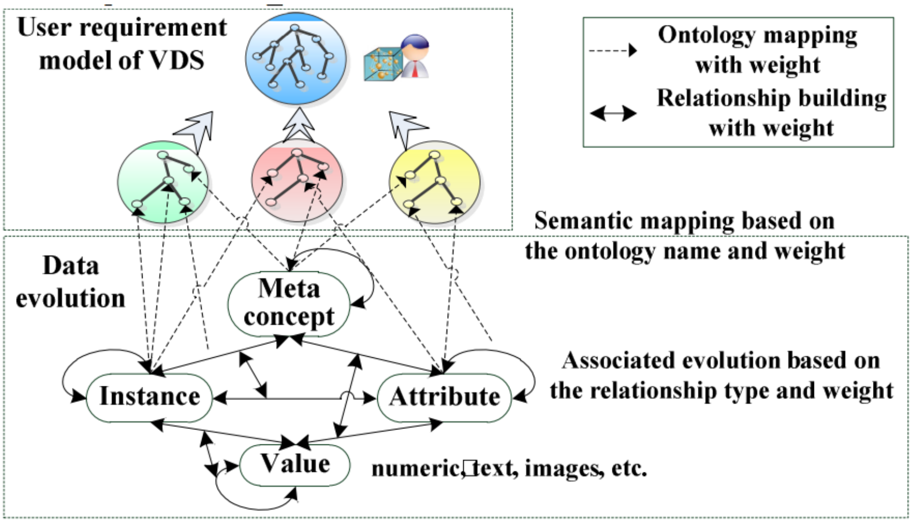

User requirement model of VDS
Ontology mapping with weight
Relationship building with weight
Semantic mapping based on the ontology name and weight
Data evolution
Meta concept
Associated evolution based on the relationship type and weight
Instance
Attribute
Value
numeric□text, images, etc.

Figure 5. The evolution process of data content in VDS.

The underlying data value such as numeric, text, images, etc. could build relationships with instance and attribute by analyzing the “value_relationship” in definition 4. Then, around the “name_ontology”, “ontomap_service” and “ontomap_weight” in the definition of Meta Concept, Instance, Attribute and Relationship, build the semantic mapping with the upper layer service by analyzing the ontology name, corresponding service and mapping weight. And then, construct the global semantic requirement model of VDS based on the local semantic association of data service. Finally get through the bottom to top contact in VDS.

Based on the above definitions, combined with the concept description about the evolution method, especially concepts 9 to 14, analyze the operation behavior about meta concept, instance, attribute, value and relationship. Figure. 6 illustrates the change process of operation behavior in the evolution cycle. When “oper_user” means a person, the evolution about this operation is explicit; and when the operation is generated by the machine automatically, the evolution about this operation is implicit. Considering the “oper_name”, “oper_origin”, “oper_time” and “oper_version”, we can get the details about change process, i.e. get the trace of data evolution in the time dimension. Meanwhile, “oper_version” could supports the version filtration in data evolution cycle, so that keeping the consistency of data version.

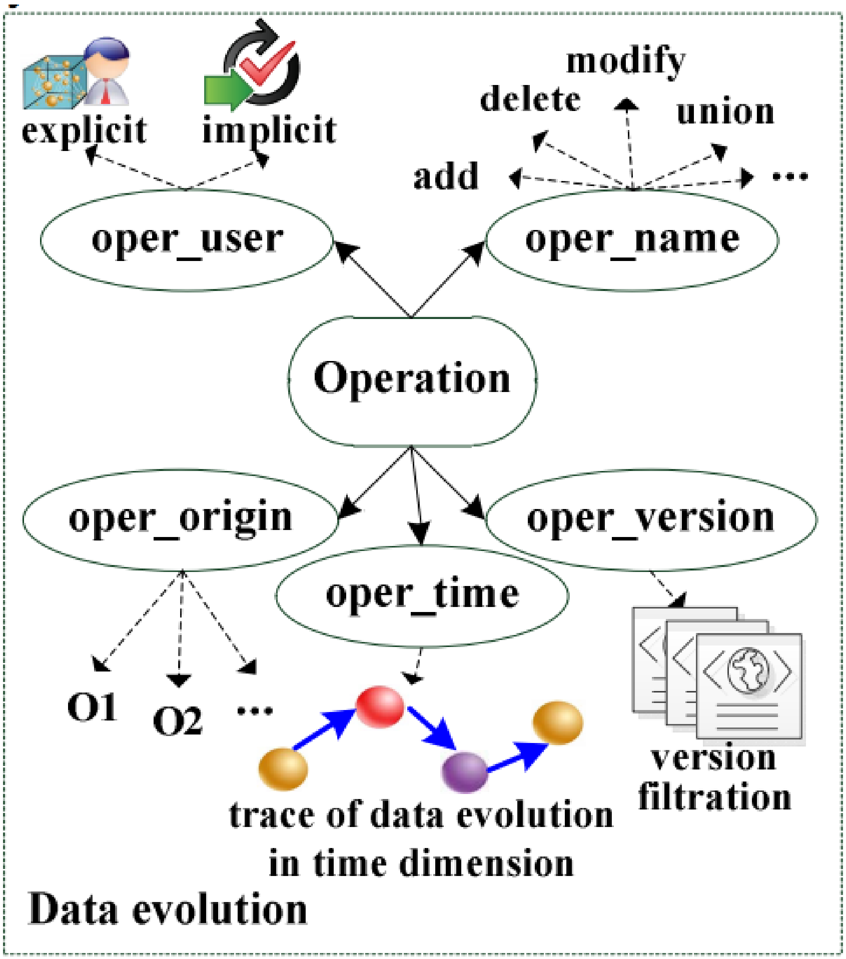


Figure 6. The change process of operation behavior in evolution cycle.

3) Automatic modeling

According to the above analysis about data content and operation behavior, combined with the related description of concepts about the analysis modeling, especially concepts 17 to 19, we can get two types of data trace. One is the dissemination trace in the space dimension by analyzing the data relational evolution. The other is the change trace in the time dimension by analyzing the data operational evolution. Along the two traces, establish the following rules to support the automatic reasoning of data evolution.

- If there exists relevance between two meta concepts, then there might exist relevance between the related instances which belong to these meta concepts; and vice versa.
- If there exists relevance between two instances, then there might exist relevance between the related attribute values which belong to these instances; and vice versa.
- If the number of data visits is large, then the importance degree of data in VDS also might be large.

Based on these rules, construct the data evolution model of VDS as the following algorithm steps.

a) Wrap and transform data: extract the key information from the distributed and heterogeneous data resources, then describe and annotate the semantic information;

b) Initially identified the data content: define the core meta concepts, attributes, instances, values and their relationships by domain experts;

c) Establish the mapping: construct the association mappings based on the anylisis of data relationship, and then map the required data to the corresponding sub VDS based on the similarity calculation of data relationships;

d) Analyze the data operation: constantly adjust the importance degree of data in VDS according to the behavior analysis, and promote the data evolution in the time dimension;

e) Improve the data evolution: continuously optimize the weights of data (i.e. the similarity at the relational level, and the importance degree at the operational level) based on the reasoning rules, thereby promote the data dissemination in the space dimension, and then realize the real-time and efficient tracking of data evolution.

In summary, the above modeling process could be able to achieve the efficient management of big data lifecycle by the automatic evolution of data in VDS. D. Cost Problem of Data Evolution.

D. Cost Problem of Data Evolution

Because of the big data management faced on the distributed storage mode, we must consider the cost problem about data evolution. Data evolution mainly contains the relational evolution and the operational evolution. So that the cost problem mainly around the following two aspects.

1) Dissemination cost

Considering the data relationships such as father, son, neighbor, similar, opposition, etc., when the data changed, the influence range in space should be controllable, and the investment by propagating these influences should be much smaller than the benefit. The investment mainly includes the task time and the storage space. The benefit mainly means the service efficiency.

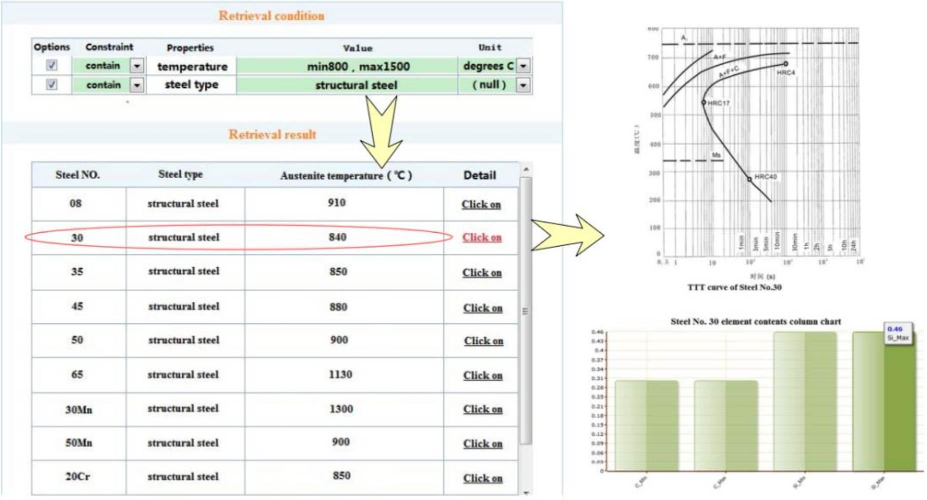
2) Change cost

Around the data operations such as addition, deletion, modification, union, merging, mixture, etc., the data evolution should develop toward the direction of more optimized data service along with the time, and also should be able to get the latest data version in real time. It is able to construct the cost model in order to support the quantitative analysis about data evolution. We will consider researching it in the future work.

V. APPLICATION CASE / Our Solution

For analyzing the applied effect of theory and method, the related approach in this paper about the modeling analysis of data evolution is used in the field of materials science for managing the domain big data. We developed a “Materials Scientific Data Sharing Service Platform (MSDSSP)” to integrate the massive, distributed and heterogeneous data sources of materials field, which based on the related technologies as VDS, big data lifecycle and data evolution. Thereby, provided the data reusing and sharing in the field of materials science, and realized the dynamic, real-time and efficient application services for the material scientists.

MSDSSP has integrated the massive data resources which could be divided into several major categories, such as the metallic materials, the organic polymer materials, the energy materials, the biomedical materials, the building materials, the road traffic materials, etc. The data nodes distributed in more than 20 research institutes which located in different regions. The data resource sets aggregated various types of relevant material information such as the grades, properties, processing, and so on. They are covered many types of materials scientific data such as the database tables, XML, images, documents, etc. This platform totally collected nearly five hundred thousand data resource items, and they keep the continually rapid growth. The relevant demands of complex association and dynamic change are increasingly significant. Considering these characteristics of materials scientific data, it is very suitable for adopting the relevant technologies about data evolution cycle to deal with the issues of big data, which is based on the modeling idea of VDS. Take the material properties retrieval as an application case to analyze and validate the data evolution process. Fig. 7 describes an application case about data evolution in material field. When we need to find the “structural steel” with the “austenite temperature” between the ranges of “800ć” and “1500ć”, through input the required retrieval condition to get the corresponding retrieval results, and then could get the more rich information about material properties by select the detail.



Retrieval condition

Options	Constraint	Properties	Value	Unit
☑	contain	temperature	min800 , max1500	degrees C
☑	contain	steel type	structural steel	(null)

Retrieval result

Steel NO.	Steel type	Austenite temperature (℃)	Detail
08	structural steel	910	Click on
30	structural steel	840	Click on
35	structural steel	850	Click on
45	structural steel	880	Click on
50	structural steel	900	Click on
65	structural steel	1130	Click on
30Mn	structural steel	1300	Click on
50Mn	structural steel	900	Click on
20Cr	structural steel	850	Click on

Figure 7. The application case of material properties retrieval about the data evolution in VDS.

During this process, the “Properties” is equivalent to a meta concept, its corresponding attribute is “austenite temperature” and “steel type”. Thereby the value of attribute “austenite temperature” with meta concept “Properties” are “min800”, “max1500” and “unitć”. While the value of attribute “steel type” with meta concept “Properties” is “structural steel”. In the result sets of query instance, there are many instances for the meta concept “Steel No.”, such as “08”, “30”, “35”, “50Mn”, etc. Then select the relationship “detail” about the instance “30” with meta concept “Steel No.”, could get the related image files as the value of instance “30”. These related images are respectively named as “TTT curve of Steel No.30” and “Steel No.30 element contents column chart”.

For the data evolution, when some data changes, such as modify the instance with meta concept “Steel No.” from “30” to “30Si”, this change could spread to the related image name based on the relationship “detail”. Then the associated values “image name” should also be changed to “TTT curve of Steel No.30Si” and “Steel No.30Si element contents column chart”.

Considering the heterogeneous data in materials field are complexly associated and constantly changed, therefore need to timely capture the related changes, and support the real-time data services according the dynamic evolution of data. Accordingly, for the impact of data evolution, VDS could be compared with the traditional database schema and the ordinary dataspace mode. Figure. 8 illustrates the comparison about data evolution among the traditional DB, the ordinary DS and VDS by considering the aspect of integrity and accuracy. Select 1000 items of materials scientific data as the specimens of data evolution, analyze the impact of data evolution which is caused by the particular changes within the same time range. It could be seen that VDS has the significant advantage in the aspects of quantity and quality about data evolution.

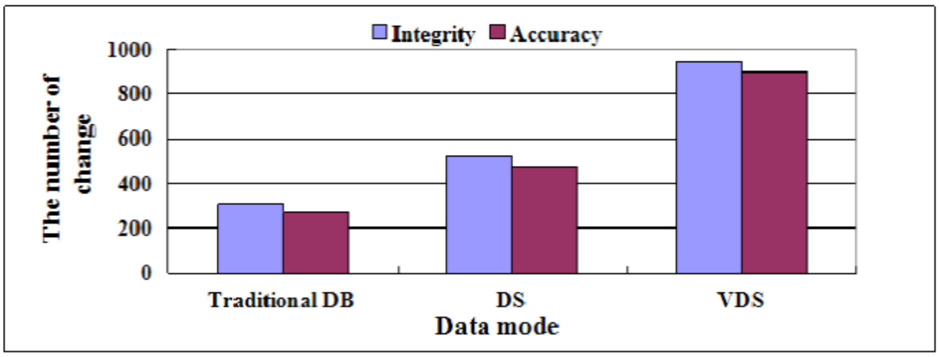


Figure 8. The comparison about data evolution among traditional database, ordinary dataspace and VDS by considering the aspect of integrity and accuracy.

Figure. 9 illustrates the comparison about data evolution among the traditional DB, the ordinary DS and VDS by considering the aspect of efficiency. Investigate the data evolution process of DB, DS and VDS, and compare the efficiency of evolution, i.e. quantitatively analyze the data evolution rate in different models. It could be seen that VDS has a very good growth trend with time. The efficiency of data evolution has reached 90% in VDS. Compared with it, the mode of DB and DS are relatively weak in the aspects of tracking efficiency about data evolution.

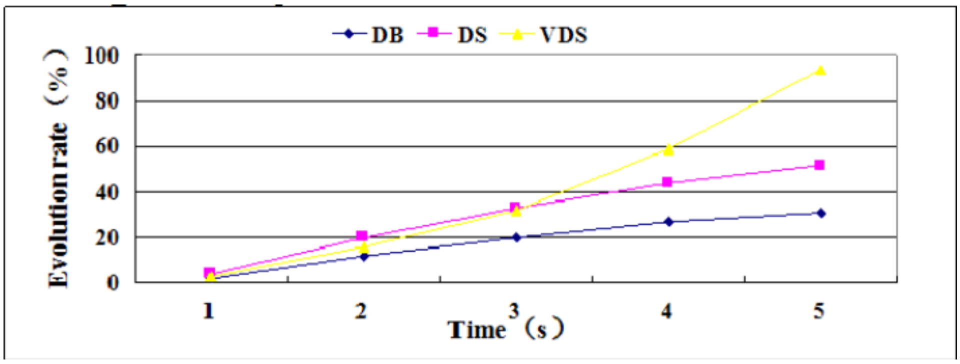


Figure 9. The comparison about data evolution among DB, DS and VDS by considering the aspect of efficiency.

In conclusion, through the analysis and description about the above specific application case, preliminarily verified the validity of modeling method about data evolution in VDS. And then realize the dynamic and real-time data services for the management of materials scientific big data.

VI. More Results and Tabels in relation to business inteliggence (Extra Chapter)

VII. CONCLUSIONS AND FUTURE WORK This paper proposed a data evolution model of virtual dataspace based on the building of big data lifecycle for realizing the dynamic, real-time and efficient application services. Through the describing of related concepts and definitions, and based on the analyzing of data evolution process in VDS, achieved the big data management in the field of materials scientific data sharing. The future work could be summarized as the following three aspects. (1) Improve the rules and algorithms to support the automatic reasoning and evolution by analyzing the data relationships which are complex and volatile. (2) For the different types of data operations, construct the detailed process cycle of data evolution by considering the different characteristics of operation behavior. (3) Research the traces of relational dissemination and operational change, and then build the cost model to carry out the quantitative evaluation for optimizing the evolution process of big data.

REFERENCES

[1] C. Lynch, “Big data: How do your data grow?,” Nature, vol. 455, Sep. 2008, pp. 28-29, doi:10.1038/455028a.

[2] D. Howe, M. Costanzo, P. Fey, T. Gojobori, L. Hannick, W. Hide, et al., “Big data: The future of biocuration,” Nature, vol. 455, Sep. 2008, pp. 47-50, doi:10.1038/455047a.

[3] Z.Y. Liu, C.J. Hu, Y. Li, and J.Y. Hu, “DSDC: a domain scientific data cloud based on virtual dataspaces,” Proceedings of the 26th IEEE International Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), Aug. 2012, pp. 2176-2182.

[4] S. Wang, H.J. Wang, X.P. Qin, and X. Zhou, “Architecting big data: challenges, studies and forecasts,” Chinese Journal of Computer, vol. 34(10), 2011, pp. 1741-1752.

[5] A. Cuzzocrea, I.Y. Song, and K.C. Davis, “Analytics over large-scale multidimensional data: the big data revolution,” Proceedings of the ACM 14th international workshop on Data Warehousing and OLAP (DOLAP’11), 2011, pp.101-104, doi:10.1145/2064676.2064695.

[6] Z.X. Qu, “Semantic Processing on Big Data,” Intelligent and Soft Computing, vol. 129, 2012, pp. 43-48.

[7] A. Simonet, G. Fedak, and M. Ripeanu, “Active Data: A Programming Model for Managing Big Data Life Cycle,” Grid'5000 Collaboration(s), 2012, pp. 1-26.

[8] M. Franklin, A. Halevy, and D. Maier, “From databases to dataspaces: a new abstraction for information management,” ACM Sigmod Record, vol. 34, 2005, pp. 27-33.

[9] L. Blunschi, J.P. Dittrich, O.R. Girard, S.K. Karakashian, and M.A.V. Salles, “A dataspace odyssey: The iMeMex personal dataspace management system,” CIDR, 2007, pp. 114-119.

[10] X.L. Dong, and A. Halevy, “A platform for personal information management and integration,” Proceedings of VLDB 2005 PhD Workshop (CIDR’2005), 2005, pp. 26-30.

[11] J. Madhavan, S.R. Jeffery, S. Cohen, X.L. Dong, D. Ko, C. Yu, et al., “Web-scale Data Integration: You can only afford to Pay As You Go,” In CIDR, 2007.

[12] X.Z. Zhang, Y.K. Li, and Y.B. Dou, “OrientSpace:Personal dataspace management prototype system,” WAMDM Technical Report, 2008.

[13] A.D. Sarma, X. Dong, and A. Halevy, “Bootstrapping pay-as-you-go data integration systems,” Proceedings of the 2008 ACM SIGMOD international conference on Management of data, 2008, pp. 861-874.

[14] C. Hedeler, K. Belhajjame, A.A.A. Fernandes, S.M. Embury, and N.W. Paton, “Dimensions of Dataspaces,” Proceedings of the 26th British National Conference on Databases (BNCOD’26), 2009, pp. 55-66.

[15] I. Elsayed, A. Muslimovic, and P. Brezany, “Intelligent Dataspaces for e-Science,” Computational Intelligrnce, Man Machine Systems and Cybernetics (CIMMACS’08), 2008, pp. 94-100.

[16] I. Elsayed, T. Ludescher, K. Schwarz, T. Feilhauer, A. Amann, and P. Brezany, “Towards Realization of Scientific Dataspaces for the Breath Gas Analysis Research Community,” IWPLS, CEUR, UK, 2009, pp. 1-8.