International Journal of Knowledge and Language Processing
Volume 7, Number 4, **2016**

www.ijklp.org
KLP International ©2016 ISSN 2191-2734
pp.37–57

Review of Data exchange system

Zhu Wenwen
Institute of Scientific and Technical Information of China
No.15 Fuxing Road, Haidian District
100083, Beijing, China
zhuww2016@istic.ac.cn

Received December 2016; revised December 2016

ABSTRACT. *With the rapid development of information technology, data exchange has become one of the key technologies to realize information sharing and efficient use. This paper summarizes the technologies used in data exchange, the standard language format and heterogeneous data extraction tools. The main technologies used include JMS, Web service and various types of middleware; the standard language format used in the information exchange process is mainly XML and JSON. Data extraction tools include Sqoop and ODI. Sqoop is an open source tool targeted to exchange data between Hadoop and relational database, and ODI is based on the concept of ELT design to provide data extraction and data conversion. By summarizing the research status of data exchange and comparing the data exchange based on JMS and Web Service, and the XML and JSON used, we get the technologies and language standard used in the process of document data exchange.*

Keywords: data exchange, JMS, Web service, JSON, XML

1. **Introduction.** In different applications, different operating environments, different development languages, there are data exchange and data sharing difficulties, the problem if the solution will cause the system is difficult or even impossible for the exchange of information, so in order to make System information can be more easily exchanged is that we need to solve the problem. Based on the existing heterogeneous platform to build a common, platform-independent, independent of the language used in the technical layer, the application of the various platforms rely on the technical layer to achieve the connection between each other and Integration, to eliminate the existing application integration problems in the program is an effective solution for heterogeneous data exchange problems.

2. **The concept and definition of data exchange.** Tian Jia [1] that the main data exchange, including data transmission and data reception. The data is sent from the server by the client to extract the original data in the user format, and then the original data of the user format is converted into the standard file by the mapping program. The communication software transfers the standard file to the designated address through the communication line according to the communication protocol; The receiving process is the inverse process of the sending process. The client uses the communication software to receive the standard file according to the communication protocol to the specified address, generate the user data format file through the process of format verification, translation and mapping, and directly use or deposit Server.

A more authoritative definition of data exchange is defined by Ronald Fagin et al. [2, 3]: data exchange can be expressed as a four-tuple (S, T , $\sum$st , $\sum$t) by IBM Alamden Research. Where S is used to represent the source pattern, T is used to represent the target pattern, $\sum$st is used to represent a set of dependencies between the source data S and the target pattern T, $\sum$t is used to represent the set of dependencies that exists in T. The process of data exchange can be described as follows: for a given instance I satisfying the source pattern, a J is found and J satisfies the following condition: J satisfies both the target format and the target relation set; the relation between I and J satisfies S And the set of dependencies between T.

Data exchange has two requirements: "synchronous" and "asynchronous". Synchronization means that the exchange process, the data exchange side to submit a data exchange request, before the exchange of results, other processes are blocked until the end of the current data exchange process. The asynchronous exchange refers to the data exchange side to submit a data exchange request, before the results of data exchange, other processes will not be blocked, when listening to the data exchange results and then continue to deal with.

3**. The main research direction of data exchange.** Summarize the existing data exchange research activities, including the following aspects：

(1) Data exchange process using different technology systems.

At present, including the use of data exchange based on JMS, data exchange based on middle-ware and data exchange based on web service. The data exchange based on JMS includes a system that utilizes a publish / subscribe function, a system that utilizes a point-to-point function, and a system that utilizes publish / subscribe and point-to-point functionality. The data exchange Based on middle-ware includes data exchange based on MSMQ, data exchange Based on MQ and MOM-based data exchange.

(2) The data exchange process utilizes systems of different linguistic formats.

On the data exchange process used in the language format include the following ①ebXML-based data exchange, ebXML is a set of norms to support modular e-commerce architecture②XML-based data exchange, which includes XML DTD.③ data exchange based on JSON, in which JSON (JavaScript Object Notation) is a lightweight data exchange language.

(3) Data exchange process to provide heterogeneous data sources of different tools.

In the data exchange process, the extraction of the source data is a key issue. Data exchange extraction tools include the use of Sqoop-based data exchange, the use of ODI (Oracle Data Integrator) data exchange system.

The following will be from these aspects of the data exchange system to explain.

4. The data exchange process using different technology systems.

4.1. Data exchange based on JMS.

4.1.1. **Introduction to related technologies.** JMS or Java Message Service (Java Message Service) Application Programming Interface is a JAVA platform for message-oriented middle-ware API, mainly used in between two applications, or is used in distributed systems to send messages, carry out Asynchronous communication. Java Message Service is a platform-independent API, the vast majority of MOM providers are to provide support for JMS. JMS is an open application programming interface developed by Sun Microsystems. It provides a general method for JAVA programs to generate, send, transfer and read message systems. The purpose of JMS is to provide a fixed interface to the clients of the messaging system which is independent of the underlying message provider.

A JMS message consists of a header, a property, and a message body. The message header generally contains the identification information and routing information of the message;Properties include the following(1) the need to use the properties of the application (2) the original message header some of the optional attributes (3) JMS Provider need to use the attributes;JMS API message body defines five message body formats, also known as the message type, TextMessage type of message body is a java.lang.String object, such as XML file content; MapMessage type of message body is a collection of name / value pairs, the value type can be any of the basic types of JAVA; BytesMessage the message body is a byte stream; StreamMessage the message body is the JAVA input and output stream; There is no message body, only the header and attributes.

4.1.2. **JMS application architecture analysis.** JMS framework of the main advantages are platform-independent; loosely coupled, including the operating state and application systems in two ways. The so-called loosely coupled operating state is the process of transmission of the message, all involved in the process (or run the thread) are loosely coupled. When the transmit direction queue transmits data, there is no need to consider whether the receiving process exists because the consumer and the producer are completely independent of each other. The loosely coupled application system mainly refers to the message technology to produce another kind of loosely coupled.Often each application is only interested in reading messages and writing messages, and they are concerned about the APIs needed to read and write messages, and JMS is a standard API.

4.1.3. **Introduction to related systems.** JMS has two standard asynchronous message delivery methods: point-to-point and publish / subscribe, providing a common way for JAVA programs to create, send, receive, and read messages from a system.

(1) Based on the publish / subscribe function

He Defu [4] and others on the difficulties of data exchange process proposed a JMS based on the publish / subscribe function of data exchange platform. The data system based

on JMS publish / subscribe is mainly the use of its theme of the release of functions, JMS Topic through the management of topics to achieve the release and subscription functions. JMS publishers are also known as news publishers, and JMS subscribers known as message consumers. When the message publisher will publish their own content which will be packaged into a message sent to the specified Topic, JMS is responsible for notifying all subscribers who subscribe to this topic. If the subscriber is online, the JMS server sends a message after establishing a connection with it. If the subscriber goes offline, the JMS server determines whether to reserve the message based on the server's subscription type. In this structure, the client will generate its own changes in the XML format of the message, sent to the message server, and other information submitted to the database server to subscribe to changes in the information to update their own database. The information distribution module records other data information which is interested in each client. When a certain data changes, the shared platform sends the updated data to the data distribution module, and then sends the updated data to the data distribution module according to the stored original data information. The data is of interest to other professional systems that implement the updating of old data.

Data exchange process is generally from the source business system through the data extraction subsystem, the automatic conversion of production standard XML file, and then enter the sending subsystem, all enter the data exchange center, and then by the data exchange center under the file subscription needs, According to the target server registration name, the message subject is sent to the target server to specify the data storage directory. Finally, the receiving subsystem loads the data into the target business system according to the set ETL rules to complete the data exchange and conversion.

(2) Based on the point-to-point function of JMS

Chen[5] based on the electronic reporting system using point-to-point asynchronous point, non-blocking message model, proposed by JMS point-to-point messaging model data exchange system. Point-to-point technology has only peer nodes at the same level, and acts as a client and a server for other nodes on the network. JMS uses a messaging mechanism for point-to-point messaging. A JMS provider is a messaging server that handles message persistence, timeout, retransmission, transaction rollback, and other services provided by JMS. In the case of the J2EE SDK, the JMS provider is part of the J2EE server program. The message producer sends the object to a queue maintained by JMS. The message consumer then accepts the message from the queue and sends an acknowledgment that the message has been received. The following figure shows the point-to-point JMS message sending process.

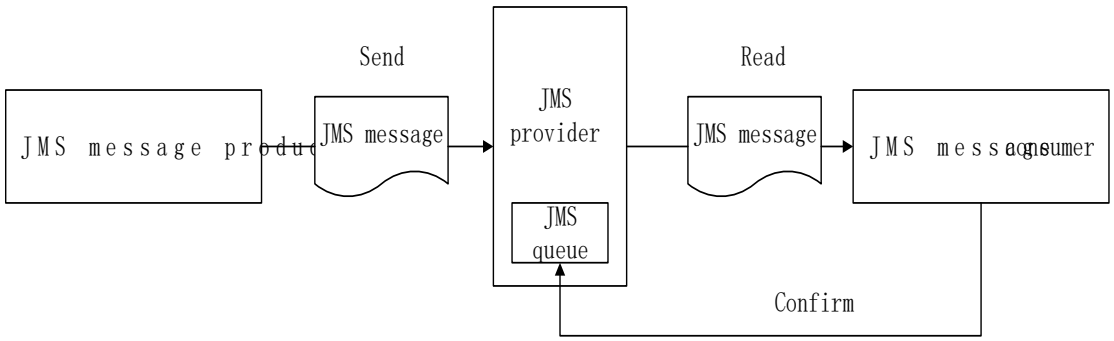

FIGURE 1[5]. JMS MESSAGE POINT-TO-POINT MESSAGING PROCESS

（3）Based on JMS publish / subscribe and point to point function

On the basis of deep research on JMS and XML technology, Zhang Xue-dong [6] proposed a new EDI model, which uses XML as the data format to realize the data conversion of heterogeneous data sources, using JMS message processing Mechanism to complete the interaction between the data process.

As the two Point-to-point and Publish / subscribe message processing models have their own strengths, so you can do two integrated system design. Point-to-point model for one-to-one messaging process, the use of this model requires prior to the establishment of a good queue between the two data sources, the message sender to send the message to the specified queue, and the message receiver from the specified queue messages. Pub / Pub model is used in one-to-many or many-to-many cases, the news publishers and message subscribers are generally anonymous, dynamic to the release of the news and subscription.

In the design of the exchange model, including the data source, data access and analysis components, XML messaging service components and JMS server. Data source is mainly responsible for the data to send and receive; data access and analysis components is to provide data encapsulation and proxy functions, will receive information from the data source, encapsulated into XML, and then sent to the XML message service components ; XML message service component is the completion of the message encapsulation and delivery, the establishment of the connection with the JMS server, XML documents will be packaged into XML messages, and according to individual needs sent to the JMS server corresponding theme or queue; JMS server is the entire The heart of the switching system, used to ensure message communication asynchrony, persistence and reliability.

4.1.4 **Summary.** JMS point-to-point model is the message producer to produce a message, the message sent to a queue, and then the message receiver and then read from the queue, once the message is read by a recipient, it will be in the This queue disappears, so a message can only be consumed by a recipient. Publish / subscribe is a one-to-many model. JMS generally focus on the exchange of information, and most of the JMS is a three-party system (consumer <-Broker <-Producer), of course, JMS can also achieve the request-response mode of communication, as long as the Consumer or Producer one broker. JMS can do asynchronous calls to completely isolate clients and service providers against traffic spikes. JMS generally does not integrate a system, but integrates a number of systems that may be involved in a message-driven environment. JMS-based data exchange

generally in the asynchronous call will have a faster speed, especially in the Java system.

4.2. **Data exchange based on middle-ware.**

4.2.1. **Data exchange based on MSMQ.** He[7] and others proposed a data exchange model based on XML Schema, MSMQ, P2P ontology, This model can fully consider the heterogeneity of each data source in the distributed environment. The newly developed system and the legacy system can coexist. At the same time, the data query and transmission between the data exchange nodes can be greatly facilitated by the way of P2P, which can greatly reduce the load capacity of the data exchange center and the network congestion.

MSMQ (Microsoft Message Queuing) technology is a technology to achieve inter-component or application-to-application communication using queuing mechanisms, which can transfer information to each other in an asynchronous, real-time manner. It has the characteristics of asynchronous communication, reliable message routing, transaction set, automatic message log, security, message routing reliability, priority, protocol independence and so on.

Message Queuing is a flexible and reliable communication mechanism that can be adapted to a variety of programs, so that developers do not need to know too much detail. It works as follows: the sender of the message you want to send out information into a container (known as the Message), and then save it to a system of public space Message Queue (Message Queue); The local or remote message receiving and processing procedures and then removed from the queue to send the message to the next step of the deal. Support for MSMQ in .Net is implemented by the System. Message namespace. For the use of MSMQ, divided into the sender and receiver programming.

4.2.2. **Data exchange based on MQ.** Yang Fan [8] proposed a data exchange model which takes XML as the data carrier and MQ as the data platform to solve the problem that the data tools between the heterogeneous databases are limited by the platform and lack of universality and low efficiency of data transmission.

MQ is between the application and the network between the intermediary software, it will be a variety of different models linked to the application to provide a simple, public, consistent programming interface, while the application and complex network technology to isolate, making Application personnel can focus on the application's own business problems, regardless of network complexity and operating environment. It is an indirect way of communication between programs, by putting messages in the queue to achieve the transmission of the message, and to ensure reliable transmission and recovery of the message**.**

In the design of the model, including the database, messaging services, and MQ Server several large modules. First of all the original data for each database data processing, mapping the original data into the corresponding XML document, and then the XML document is encapsulated into the message, and through the MQ to transfer, to achieve heterogeneous database Between the data exchange. The core functions in this model are as follows:

(1) message processing module: including MQ Server and messaging services

MQ Server: MQ front-end, which is mainly contains the MQ queue manager and specific message queue, and the realization of the MQI interface, which is responsible for the need to transfer the message to the target system MQ Server, and is also responsible for MQ Servers on other systems receive messages.

Message Service: to achieve the connection with the MQ Server, the converted XML documents to MQ messages to the package, and can be removed from the MQ message XML documents.

(2) data processing module: the main is to receive data from various databases in the XML document structure and the establishment of the corresponding mapping between the databases, but also have to be able to parse XML documents, the data content mapped to the corresponding database in.

4.2.3. **MOM-based data exchange.** Zhang Qian [9] and others based on the message middleware (MOM) technology in the message transmission process of the asynchronous and reliability support, design and implementation of the MOM-based web services data exchange system.

MOM is a kind of middleware, based on message queue storage and forwarding mechanism to support asynchronous transfer of data between applications, that is, each application does not communicate directly with each other, but with the MOM as an intermediary communication. MOM enables applications to send requests to applications that are not working or are unreachable, providing a store mediation staging message to ensure that messages are sent as soon as the network is connected or the receiving application begins processing. The function of MOM is to control and manage an integrated system, so that the whole workflow is completed by several messages, queues and queues composition.

System mainly by the system configuration management, web service interface, queue management and log management of several parts. System configuration management is mainly used to complete the need to exchange the data publisher or the data subscription side of the registration; Web service interface is mainly used to complete the exchange of information to be transmitted between the two sides of the XML format file encapsulation and analysis; queue management, including management queue, Task and thread pool three parts, the management of the queue is responsible for the release and subscription of the task queue and data queue management, management tasks when the data exchange is the time to arrive, send the corresponding message notification and task manager, thread pool It is mainly used to be responsible for data exchange; log management is to record the relevant data exchange operation log and to provide the query service.

The data exchange system for the B / S mode, the choice of open-source Tomcat software for the web server, the choice of XML for the exchange of data in a unified format, use Hypertext Transfer Protocol (HTTP) for data exchange protocol. Different from the web service request mode, the system uses publish / subscribe mode for data exchange. Taking into account the actual exchange mostly for basic data, the demand for real-time is not high, the system uses the timing of data exchange timing method, according to the exchange system to control the data exchange time or time point, thus greatly improving

the data interaction effectiveness, Waste of resources.

4.2.4. **Summary.** Professional middleware products can provide stable and reliable file transfer, simple and efficient application integration, comprehensive management function system, in order to solve the existing data island and system isolation problems, to provide professional, advanced, high efficiency, low cost solution.

4.3 **Web-based data exchange.**

4.3.1. **Introduction to related technologies.** Web services use SOA (Service-Oriented Architecture) architecture. The architecture consists of three participants and three basic operations. The three participants are Service Provider, Service Requester and Service Registry. The three basic operations are Publish, Find and Bind. The service provider publishes its services to a directory on the service proxy; when the service requester needs to invoke the service. It first to the directory provided by the service agent to search the service, get how to call the service information; and then according to the information to call service providers to publish the service. In the Web Service architecture, the use of WSDL to describe the service, UDDI to publish, find services, SOAP used to perform service calls. WSFL will be scattered, single-function Web Service organized into a complex organic whole. When the service is called. It first to the service agent to provide the directory up to search the service, get how to call the service information; and then according to the information to call the service provider service. In the Web Service architecture, the use of WSDL to describe the service, UDDI to publish, find services, SOAP used to perform service calls. WSFL will be scattered, single-function Web Service organized into a complex organic whole.

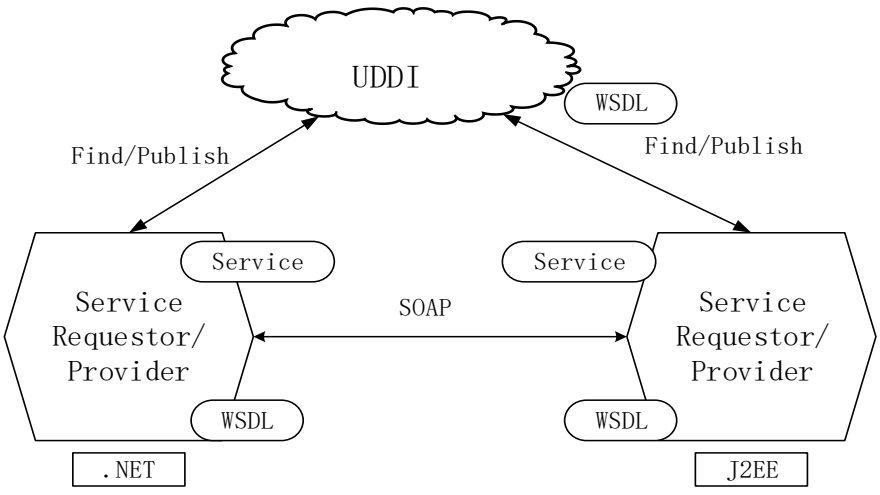

FIGURE 2. WEB SERVICE ARCHITECTURE

SOAP (Simple Object Access Protoc01): is a simple protocol for exchanging information in a distributed environment and is an XML-based protocol.

WSDL (Web Service Description Language): is a logical interface that describes the application system to describe the application of external services, and so that all

applications can be between the WSDL to communicate and integrate. It is a document composed of XML, describes the realization of procedures provided by the prototype function, for other used to wither.

UDDI (Universal Description Discovery and Integration): is used to register in a structured way, management information and related service information. It provides mechanisms for publishing, searching, and using services. It provides detailed information about external services in terms of providers, service descriptions, and service bindings.

4.3.2. **Related system implementation.** Wang Yanmin [10] and others for data exchange in the solution of information access problems exist heterogeneous, proposed on the basis of XML technology based on ontology and web service data exchange platform.

The system framework mainly includes switching agent, web service interface, mode management and transformation rule management. The exchange agent mainly implements the shielding of the heterogeneous data source, transforms the data from the data source and the XML document, generates the corresponding XML Schema, and provides the unified web service interface and the data exchange pool for the cross-platform interaction. Web service interface mainly provides data pattern access interface, data service interface and data interaction interface. Schema management is mainly responsible for the preservation and management of the database in all the data source file XML model, in the exchange of time for pattern matching. Transformation rule management is the mapping between data patterns that are primarily used to establish a data distribution share and the data patterns required by the recipients of the data.

The entire data flow may be described as follows: the application system 1 sends a data request to the data exchange center, which forms the translation scheme and forwards the data request to the application system 2, and then applies the conversion scheme to the data sent back from the system 2 for data processing and The results are sent back to the application system 1 via data center processing. When the application system enters the exchange system for the first time, the XML Schema of the data source is generated by the exchange agent and the XML Schema is submitted to the data exchange center; the data exchange center marks the XML Schema by using the ontology to form the semantic information mode file, And then through the conversion program generator to form a final program, and saved in the conversion rule base. When exchanging data with other application systems again, the data to be exchanged is converted into an XML document by a switching agent and submitted to a data exchange center when the data is exchanged again with another application system; the data exchange center converts the XML document into a conversion rule base by using a conversion scheme generator the XML document that the target system recognizes.

4.4. **Summary.** Because of its interoperability, universality, ease of use and high scalability, Web services are the popular way to connect heterogeneous applications to exchange data, but there are several problems: (1) cannot support complex Data (2) data transmission and data exchange engine over-coupling (3) the use of unreliable HTTP protocol, there will be unreliable data transmission. Therefore, two kinds of solutions are proposed for the reliability of web services. One is to guarantee reliable transmission by extending the

header message, such as the WS-Reliablity standard proposed by the Structured Information Standards Organization. The essence of this standard is to add the acknowledgment and retransmission mechanism in the SOAP message header to ensure that the message is delivered in one time and in order. This approach solves the reliability problem, but it increases the complexity of web services application layer processing logic; the second type of solution is to use a reliable transport protocol instead of the underlying unreliable web services HTTP protocol, such as the use of JMS (MOM) to ensure the reliable transmission of data, so that the web services framework can be divided into two parts, the upper layer (the upper layer) and the upper layer (the lower layer) To provide services, the lower to provide reliable transmission.

Web services treat everything as a service, which can be dynamically discovered, organized, and reused on a network through a messaging mechanism. Web services are encapsulated into services described by WSDL, which shields the complexity of business logic, the diversity of technologies and the heterogeneity of development platforms. Web services can be implemented based on HTTP, SMTP, JMS and other protocols, can be asynchronous or synchronous. Web Service interface through the security management to achieve a trusted web service call.

Web service implementation are mainly the following three: (1) Remote Procedure Call (RPC): Web services to provide a distributed function or method interface for user calls, which is a more traditional way. Typically, RPC interfaces are defined in the WSDL. (2) service-oriented framework (SOA): The more popular is from the concept of service-oriented framework (SOA) to build web services. In a service-oriented architecture, communication is message-driven, not an action (method invocation). This web service is also a message-oriented service. SOA-style Web services are recognized and supported by most major software vendors and industry experts. As the biggest difference with the RPC approach, SOA approach is more concerned about how to connect to the service rather than to specify the details of an implementation. The WSDL defines the necessary content for the contact service. SOA is not a language, nor a specific technology, it is a new software architecture model. SOA is the way to construct an application for distributed computing. It sends the application as a function to the end user or other service as a function. (3) Representational State Transfer (REST): These web services focus on the interaction of stable resources, not messages or actions. REST service definition: is a design and development approach for network applications, can reduce the complexity of development, improve system scalability. The REST architecture follows the CRUD principle, which requires only four behaviors for the resource: Create, Read, Update, Delete, to complete its operation and processing.

5. **The data exchange process using different language formats of the system.**

5.1. **ebXML-based data exchange.**

ebXML [11] (Electronic Business using eXtensible Markup Language) is a set of norms to support modular e-business architecture, which supports a global electronic market, so that any size of the enterprise through the exchange of XML specification information, To contact and processing data. ebXML is a global initiative developed and used by UN / CEFACT (the Center for Promotion and

E-Commerce) and OASIS (Organization for Standardization of Information Standards). Its goal is to enable businesses of any size to conduct e-business with anyone Business.

The ebXML Modular Architecture is comprised of five major modules including messaging, BPSS, CPP & CPA, registry and knowledge base, and core components.The ebXML message uses the SOAP specification; ebXML differs from other XML in that it emphasizes more on business process specifications, captures and uses standard formats to represent the business data flow between trade gray boards using modeling languages such as UML and charting tools, ebXML also provides a registry component that includes industry processes, messaging, and the ability to use e-mail services for e-commerce, e-commerce, e-commerce, e-commerce, e-commerce, e-commerce and e-commerce. Define transaction vocabularies for trading data between trading partners.

Because traditional data exchange system lacks universality and expansibility, all parties involved in data exchange need to strictly follow the same rules for data encapsulation and analysis. Cao et al. [12] proposed an ebXML-based data exchange system Architecture, through the construction of XML standards in line with the business willow into a standard, you cannot consider the underlying data exchange details, to achieve seamless data connection.

Business resource management subsystem is divided into two parts, namely the daily system management and log management functions, as well as ebXML registry and knowledge base functions. The ebXMLrr (ebXML Registry / Repostory) is chosen as the component of this subsystem to realize the function of registration and knowledge base.

The data exchange service subsystem consists of three parts. The message service module can realize the reliable receiving and sending of the message. The whole message processing standard and mechanism must conform to the ebXML standard. Here we choose HermesMSH development software, it is a sub-project in freebxml organization, in line with 0ASIS ebXML message service standards, can be used as support ebXML standard message processing middleware, use it to provide client development kit SDK can achieve message processing service.

The management control subsystem implements the daily management of data exchange services, including service control, log viewing and policy management. The subsystem uses B / S architecture, sub-presentation layer, business layer and data integration layer, the presentation layer using Struts plus common control architecture.

Message Format: We have established the corresponding specification for message passing among the subsystems. The message format follows SOAP and SOAP attachment standard. The message format is based on SOAP + MIME, and each message consists of several MIME parts

ebXML builds standard business process specifications to revolutionize the way data is exchanged. By following ebXML's business process specification and the basic idea of business collaboration agreements, we can integrate excellent open source by following the process of ebXML application development. Software, designed to meet the needs of individual data exchange system.

5.2. **XML-based data exchange.** XML (The Extensible Markup Language) Like HTML, XML comes from SGML (Standard Generalized Markup Language), and is one of its own. XML is a simple, flexible text format, a set of rules for semantic markup that divides a document into many parts and identifies them[13].

XML can be content and form of separation, making data reuse, so XML as a common data format, you can handle text, images and sound and other formats of data, and can be extended by the user to deal with any special type of data.

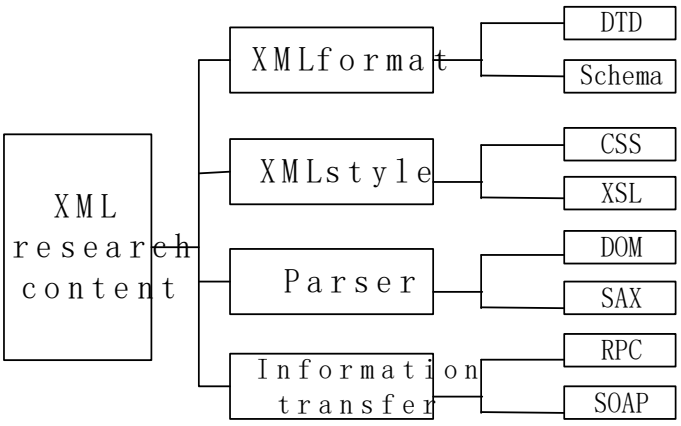

FIGURE 3[14]. XML RESEARCH CONTENT

For a batch of XML data, the application processes and displays the XML document, the document type specification (DTD / Schema), and the style (CSS / XSL). First, the document type describes the structure, content, and limitations of the XML document. The XML document then stores the data according to the format of the DTD / Schema archive, and finally the XML document is published according to the style description[14].

There are two ways to access XML documents programmatically: the Document Object Model (DOM) based on tree representation and the Simple XML Application Programming Interface (SAX) based on events. DOM is a data exchange format XML as a DOM object, you need to read the XML file into the entire memory; SAX do not need to read the entire document can be resolved on the content of processing, is a step-by-step analysis. The program can also terminate the parsing at any time. In this way, a large document can be gradually, a little bit of the show, so SAX suitable for large-scale analysis. RPC and SOAP are two commonly used XML-based data exchange protocol, RPC does not require mapping, packaging, just add a little constraint on the XML document format, you can directly transfer XML documents; SOAP goal is to create a XML Lightweight protocol for exchanging structured and defined type data in a distributed loosely coupled network environment.

5.2.1. **Data exchange based on XML DTD.** DTD is used to define the logical structure of XML documents, which defines the elements of XML documents, elements of the property and the relationship between elements and elements. So the same XML document can be interpreted by different DTDs to give the actual meaning to the data in the XML document, so that different organizations in the same field can understand each other's XML data, which is the difference between the different systems Data exchange provides the objective

conditions, but also makes online data exchange simple.

He Guohui [15] and others combined with the characteristics of XML DTD, it has become a data exchange file capability, combined with the data exchange process, put forward the XML file as a data exchange platform approach.

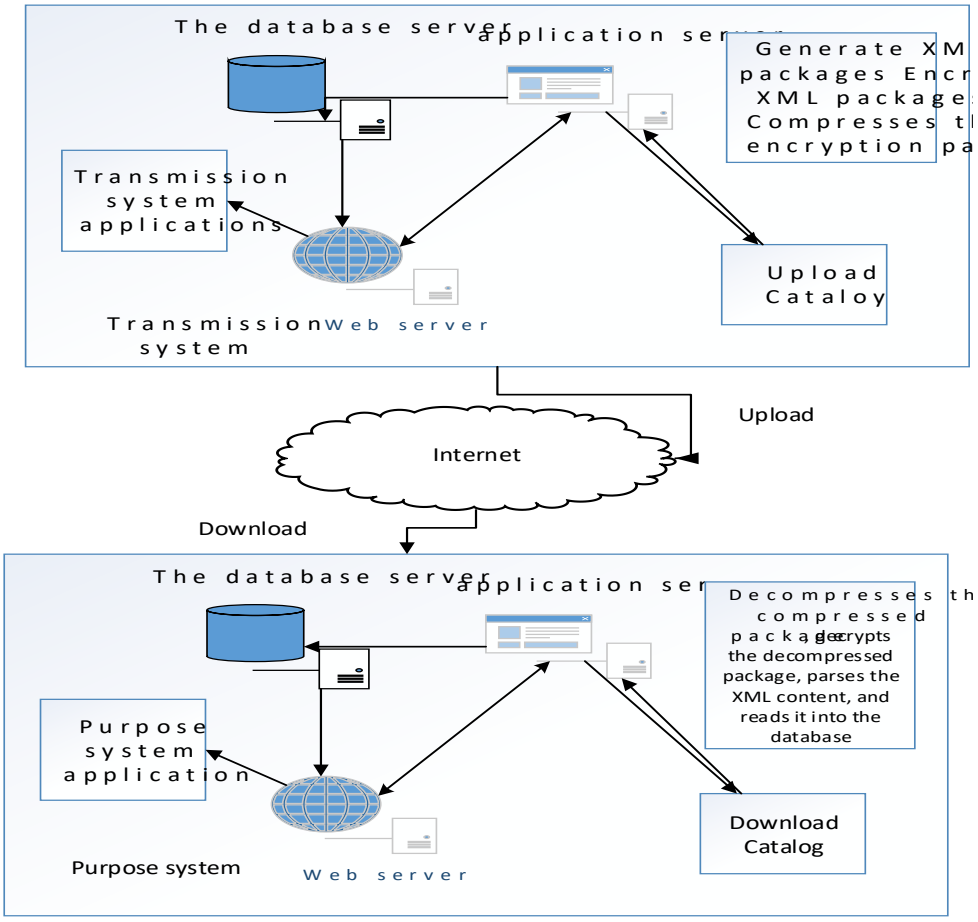

FIGURE 4[15]. THE OVERALL STRUCTURE OF THE SYSTEM

The system works as follows: When the system needs to send the relevant information in the database to other systems through the XML document, it can generate the XML document by clicking or activating the function automatically by the system, encrypting the XML document, Package compression and stored to the upload directory function, another component of the way through the query to constantly check the upload directory of the file, when found to be required to upload files, automatically upload the file to the specified location. While the data receiving part of the work is to send from the transmission system to analyze the data packets.

Considering the complexity of the system structure, in order to facilitate the expansion and maintenance functions, the system uses three-tier or multi-layer structure, configure the web server, application server and database server. Web server will be responsible for running the browser-side display of the application logic, the database server is responsible for the connection with the database, While the application server is between the client and

the database server, a class of servers, also known as the intermediate server, will be responsible for Deal with the business logic.

Based on SpringMVC architecture approach, the establishment of the database used Hibernate mapping. Hibernate is a pure Java object-relational mapping and persistence framework that allows you to map plain Java objects to relational database tables through XML configuration files. Hibernate can save a lot of project development time. Since the entire JDBC layer is managed by this framework, it means that the application's data access layer will reside above Hibernate and can be abstracted entirely from the underlying data model.

5.2.2 **Data Exchange Based on XML Schema.** Schema is a language used to describe and standardize the logical structure of XML documents, its biggest role is to verify the correctness of the logical structure of XML documents. In addition, Schema supports namespaces, built-in simple and complex data types, and support for custom data types. Schema output in two forms: Schema XML file form and OutputStream form. Output Schema in the form of an Outpotstream so that other programs can use it.

XML Schema format and XML DTD format has a very significant difference, XML Schema is actually an application of XML.That is, the XML Schema format is exactly the same as the XML format, and as a subset of the SGML DTD, the XML DTD has a completely different format from XML. The DTD uses the EBNF syntax.

Li Zonghua [16] proposed a lightweight data integration method based on XML Schema for data warehouse and federated database to solve the complexity and cost of data integration.

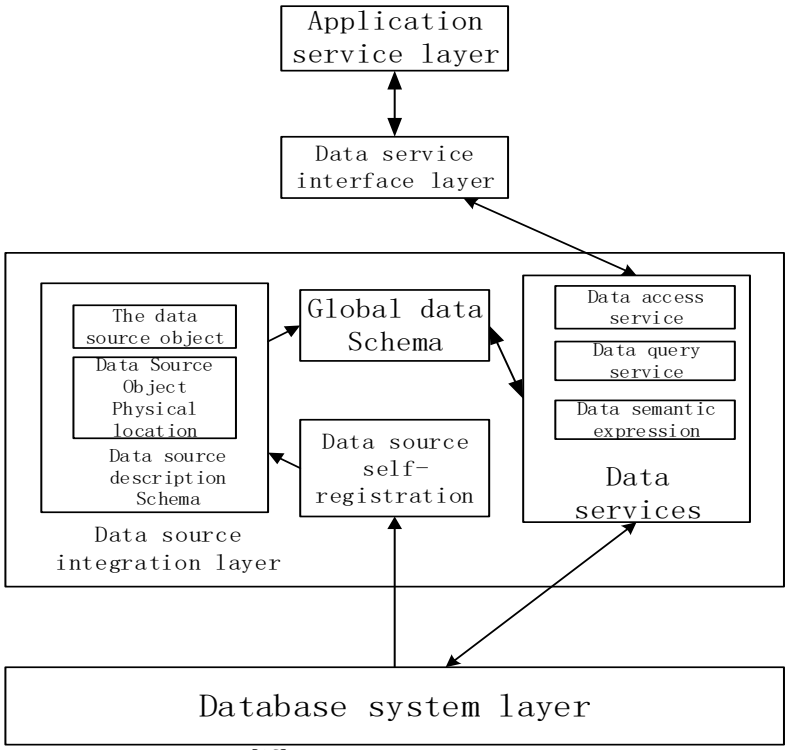

FIGURE 5[16]. DATA INTEGRATION FRAMEWORK

The data integration framework focuses on the data integration layer and the data service interface layer. Among them, the data integration layer includes data source self-registration, data source object and data source object physical location description Schema design, global schema design and data service. The data service interface layer uses Remote Method Invocation (RMI) technology in Java to create remote method invocations.

The integration framework includes a schema describing data source objects and physical locations, a global data object schema, a distributed query strategy, and a data service interface.

5.3. **JSON-based data exchange.** JSON (JavaScript Object Notation) is a lightweight data exchange language, text-based, and easy to read, but also facilitate the machine to parse and generate. JSON uses a text format that is completely independent of the programming language, but also uses a Similar to C language (including C, C ++, C #, java, JavaScript, Python, etc.) that make JSON the ideal data exchange language.

JSON format data is designed for the browser-side language JavaScript designed to generate the JSON object can be directly parsed, with a strong versatility. Therefore, through the analysis of JSON, the use of its value to the way the performance of the data a bit, the basic structure of XML data analysis, a JSON format with the corresponding conversion rules. Through this rule will be converted to XML format data JSON format, and JSON data can be directly through the browser-side JavaScript language resolution.

JSON builds with two structures [17]:

A: A collection of name / value pairs. Different languages, which are understood as objects, records, dictionaries, hash tables, keys lists, associative arrays.

B: an ordered list of values, in most languages, he was understood as an array.

In order to solve the problem of low efficiency and commonality of traditional XML-based data exchange platform, Zhang et al. [18] proposed a data exchange model of "center + agent" based on JSON. Using JSON as the bus technology, data exchange center as the architecture center.

The data exchange center (DEC) is used as the architecture center, and the data exchange agent node (DEAN) provides the proxy service transaction design structure. The entire architecture is a star structure, DEC in the central location, it is the center of data exchange, it through a standard web service interface for each DEAN to provide services. Data exchange process is that each application has DEAN as a proxy interface and DEC for the exchange of messages and data, all messages and data exchange are JSON information flow.

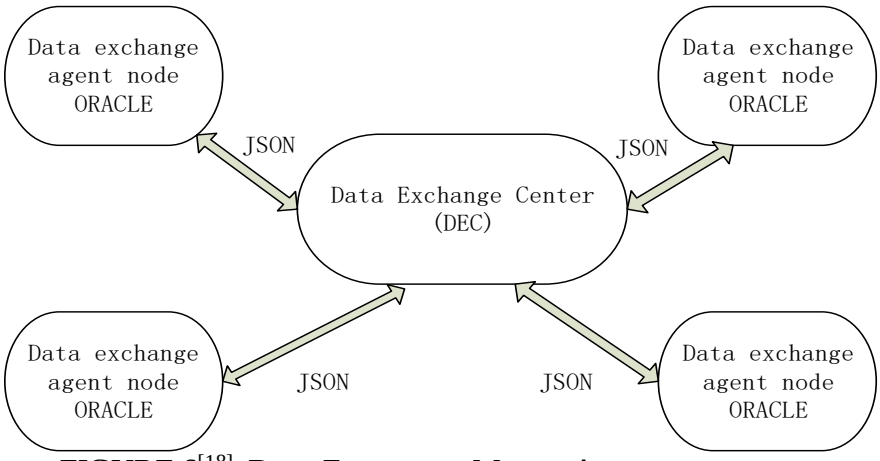

FIGURE 6[18]. DATA EXCHANGE MODEL ARCHITECTURE

The underlying implementation of the entire data exchange is transparent to each application node, which is low in coupling and easily scalable to a hierarchical snowflake structure, constructed as a multi-level data switching center architecture to support a wider range of wide area solutions. When adding a database type, only need to establish a corresponding proxy node can be connected to the DEC.

5.4. **Summary.** Serialization of JSON the process of converting the state of an object to a format that can be persisted or transferred. The opposite of serialization is deserialization, which converts a stream to an object. This combination of the two processes makes it easy to store and transfer data [19].

Because JSON's concise data description format, its data transmission overhead is smaller than the XML transmission format, and the reduction of data transmission cost will inevitably bring about the improvement of data transmission efficiency. Gao Jing [20] and others made on the JSON data transmission efficiency study by experiment, the client, from the server-side transmission of JSON or XML data will be deserialized to be able to access the data, and then displayed on the client page. Among them, XML is based on the DOM tree structure, deserialization of XML need to consider the parent node and child nodes, which will increase the difficulty of deserialization, and JSON only through the JavaScript language eval () function can JSON data Serialized as a JavaScript object. Therefore, the use of JSON format data greatly reduces the deserialization of the redundancy, making development efficiency. If combined with AJAX technology will make real-time interface update effect is remarkable, can better improve the user experience.

But the XML as the transmission of the advantages of uniform format, standard, and easy to interact with other systems, data sharing more convenient. XML format and popularization of the popular and widely used, with the versatility.

By comparing the DTD and Schema, we can see Schema has obvious advantages, roughly as follows:

(1) XML Schema itself is also a kind of XML, so many XML editing tools, API development kit, XML syntax parser can be used directly to XML Schema, without modification.

(2) As an application of XML, XML Schema should inherit the self-description and

extensibility of XML, which makes XML Schema readable and flexible.
(3) Since the format is exactly the same as XML, XML Schema can be handled in the same way as XML. It can also be stored in the same way as the XML document it describes.
(4) XML Schema and XML format consistency, making XML data exchange for the application system can also facilitate the exchange of models;
(5) XML has a very high legitimacy requirement, XML DTD description of the XML, often used to verify the legitimacy of XML is a foundation, but XML DTD itself lacks the legitimacy of a better authentication mechanism, must be independent. XML Schema is different, it and XML have the same legitimacy verification mechanism.
(6) XML DTDs have obvious shortcomings in the description of relational data. For example, the limited data types of XML DTD cannot complete one-to-one mapping of data types, and cannot describe most data rules. XML Schema provides more built-in data types, and supports the user's data type extension, basically to meet the relational schema in the data description needs, which can be used as XML Schema XML DTD is more suitable than describing a relational data main reason.
(7) Schema provides support for namespaces. Support for namespaces allows you to mix names from different sources.

Through some comparison, we can see that XML Schema as a powerful standard, XML DTD than the more expressive, to meet the needs of more different areas. So we can use XML Schema as the standard format for data exchange.

6. **Data exchange process Different data sources to provide heterogeneous tools.**

6.1. **Sqoop-based data exchange system.** Sqoop is an open source tool that was developed to exchange data between Hadoop and relational databases. You can through Sqoop the data from the database (such as mysql, oracle) into hdfs; can also export data from the hdfs relational database. Sqoop through Hadoop's MapReduce import and export, thus providing a high parallel performance and good fault tolerance. It is based on MapReduce for data processing, so Sqoop must rely on Hadoop cluster environment.

Hadoop is an open source distributed service platform, because of its high reliability, high efficiency, high fault tolerance and strong horizontal scalability, it is widely used in the field of large data, is now popular in the large data industry. Yu Jinliang [21] and others on the use of the advantages of Hadoop itself, the data storage, analysis, processing purposes, and in the data transmission process to take into account the transmission efficiency and data quality; proposed data transfer tool Sqoop relational database data Import Hadoop platform distributed file system (HDFS), non-relational database (HBase), relational data warehouse (Hive) stored in order to achieve data exchange.

Sqoop through the introduction of Sqoop Server (specific server for Tomcat), for each database connection (Connector) for centralized management. It can also be accessed in a variety of ways, through the REST API, Java API, web UI and CLI console, etc. to control the process of data exchange. In terms of security, the command-line control of the way in the relationship between the database and Hadoop data exchange, the command will have an interactive interface, enter the development of relational database user name and password will not be seen.

In the data exchange, the system can read the relational database data, the data into the Hadoop cluster HDFS, HBase, Hive, in order to achieve the relational database and Hadoop data exchange between purposes. HDFS is Hadoop distributed file system, with high fault tolerance features, can be deployed in low-cost machines, and to maintain high reliability of the data. Hive is a data warehouse built on Hadoop that can be seen as a user programming interface that does not store and calculate data itself, relying on HDFS to store data, MadRduce to process the data. HBase is a non-relational (NoSQL) database that runs on Hadoop and is a distributed, extensible database. The Hadoop-based Distributed File System (HDFS), the most basic data storage unit, You can view this data using the HDFS client, and you can also work with HBase through the Hadoop Distributed Computing Framework MapReduce.

6.2 .Data exchange based on ODI. ODI is Oracle's acquisition of Sunopsis in October 2006, the integration of Sunopsis Active Integration Platform launched a data integration tool, is based on the concept of ELT design data extraction / data conversion tool. It is different from the traditional ETI tools: ODI supports heterogeneous data exchange between the data integration, not just limited to the exchange of ORACLE database integration, the implementation of data exchange provides a design method to support complex and real-time Of data integration, to achieve seamless integration between systems.

ODI is part of the Oracle Fusion Middleware product family, which addresses data integration requirements in heterogeneous environments. It is a Java-based application that can use a database to perform collection-based data integration tasks, or extend the functionality to multiple database platforms and Oracle databases. ODI can adapt to different, a variety of data sources, flexible and effective completion of data extraction / conversion / loading process, are based on its knowledge model system. ODI has more than 100 knowledge modules (Knowledge Modules) similar to plug-ins in the program, support hot-swappable, an increase of ODI flexibility and scalability. ODI data integration tasks abstracted out six components, the main application of the project is LKM and IKM two knowledge modules. The load LKM is used to extract data from the data source, and the integrated IKM is used to convert the data in the Staging Area to the target table, generating the corresponding translation SQL based on the target database.

The system data exchange between the company's higher data requirements and larger, so Xu Canfei [22] and others through the middleware interface program, using ODI server as an inter-system data exchange platform for integrated tools to achieve a number of independent and stable information systems Running at the same time, to ensure a seamless connection between multiple information systems.

The use of ODI for all business systems in the enterprise data integration, cleaning, switching and synchronization using the middleware model. It has the advantage of shielding the underlying data complexity, so that developers face a simple and unified development environment to integrate data sources, greatly reducing the technical burden.

Using ODI to achieve enterprise information management data exchange platform, the system between the successful completion of the transaction processing conditions to

ensure the data between the various systems of efficient and accurate exchange. Another great application lies in its scalability, which not only meets the existing information system data exchange, but also to meet the new application system and the future data exchange needs.

7. **Conclusions.** The main advantage of XML as the main transmission format of web information transmission is ① unified format, conform to the standard; ② easy to interact with other systems, data sharing is more convenient. The purpose of Web Services technology is to solve the network environment of application sharing, and also to XML as a data format, which for enterprise resources from the internal sharing to the network share provides a strong technical support. So for enterprises to achieve distributed application system to solve the two internal and external problems, as long as the XML-based web services and enterprise application development can be combined together to solve.

The biggest advantage of XML compared to JSON is its versatility. JSON works in JavaScript at home and can store JavaScript composite objects, but it is not easy to make syntactic-qualified JavaScript code on the server side, System, the server and the client have different developers, they must consult the object format, which is very prone to error.

In addition, JSON is a very good data format, but it is just a data format. XML is a very powerful language, rather than just a simple data format. XML has at least the following important features over JSON and other simple data formats: (1) XPath, in order to get the publication year from the document, etc., simply sends a simple XPath request, but there must be an XPath Processor to resolve the request and return, which is JSON can not do (2) Attributes and Namespaces, metadata can be added to the XML data. Saving data in an element will greatly benefit organizational and structured information. The advantage of this technique is more apparent when multiple applications use the same XML document. (3) XML Schema, when an XML document created on a machine, and then made a number of other computer changes, and then spread to other computer use, you must ensure that the document structure has not been destroyed by the middle operation, so We can create a description document XML Schema, and the main document with the preservation. Each time the main document before the operation, need to check through the Schema file its correctness, which is an integrated production process testing. (4) XSL, in fact, we can not use any Java / Ruby and other code can be completed XML document changes. In this process, you only need to create an XSL transformation document and apply it to the original XML, you can get a new XML. XSL language (purely functional language) is designed for hierarchical data manipulation design, than JAVA or any other object-oriented / process language are more suitable for this task. XSL can be used to convert XML into any form, including plain text and HTML.

The light weight difference between JSON and XML is that JSON provides only an overall parsing scheme that works well with less data; XML provides a step-by-step analysis of large-scale data, which the scheme is well suited for handling large amounts of data. So when dealing with large amounts of data, we can choose to use XML as a data format standard, when dealing with small amounts of data, the advantages of JSON to fully show up.

Web services focus on remote service invocation, in most cases the direct interaction between the two systems (Consumer <-> Producer), usually synchronous calls, the need for complex object conversion. Web services are network-based, distributed, modular components that perform specific tasks and comply with specific technical specifications that enable web services to interoperate with other compatible components. Web service is essentially just a service component, although it uses a standard protocol, but it is closely related with the application service. SOA is a service-oriented, and this type of service is the result of neglect and technology-related things, and ultimately provide the service interface.

When we need to exchange information about documents, patents, scientific reports, etc., we need to extract the data from the data sources and exchange them to shield the heterogeneous data sources from the format differences, generate the corresponding XML Schema, Web service technology, and provide a unified web service interface and data exchange center for interaction. The data exchange center mainly includes identity authentication, XML parsing module, XML encapsulation module, traffic management, log management and so on.Through which to achieve the final data exchange process.

Acknowledgment. This work is supported by National Digital Composite Publication System Project (XWCB-ZDGC-FHCB/29). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] Tian jia, Design of electronic commerce data exchange platform based on XML, *Automation& Instrumentation*, no.4, pp.37-39, 2016.

[2] Fagin R, Kolaitis P G, Miller R J, et al. Data Exchange: Semantics and Query Answering.*Database Theory — ICDT 2003*. Springer Berlin Heidelberg, pp.207-224, 2003.

[3] Fagin R, Kolaitis P G, Popa L. Data exchange: getting to the core. *Acm Transactions on Database Systems*, vol.30, no.1, pp. 20-101, 2004.

[4] HE Defu,SU Xisheng,FANG Jingshuai.Data Exchange Platform Based on JMS.*Computer and Digital Engineering*, vol.44, no.8, pp. 1518-1522, 2016.

[5] CHEN Shi, The Data Exchange Mechanism of Electronic Declaration System Based on JMS.*OA'2007 Office Automation Symposium*.2007.

[6] Zhang Xuedong, Design of EDI based on JMS and XML, *COMPUTER DEVELOPMENT&APPLICATIONS*, vol.19, no.4, pp.2-3, 2006.

[7] HE Wei,XU Dong-ping,ZHANG Bian,Research On the MSMQ-based Data Exchange System.*COMPUTER KNOWLEDGE AND TECHNOLOGY*, vol.4, no.20, pp.13+133, 2007.

[8] YANG Fan, Technology of Data Exchange Between Heterogeneous Databases Based on XML and MQ.*Computer and Modernization*, no.6, pp.147-150, 2013.

[9] Zhang Qian, LI Minghao.Design and Application of Data Exchange System Based on Web Services and and MOM.*Command Information System And Technology*, vol.6, no.1, pp.70-74, 2015.

[10] WANG Yan-min,XIE Qiang,DING Qiu-lin.Data Exchange Platform Based on Ontology and Web Services.*COMPUTER TECHNOLOGY AND DEVELOPMENT*, vol.20, no.5, pp.112-116, 2010.

[11] Wang Xilin,《*E-Business Standardization Guide*》.China Standard Press, 2004.

[12] CAO Cheng,SHU Jian,CAI Ke,JIANG Wei,An ebXML-based Data Exchange System Architecture.*JOURNAL OF JIANGXI NORMAL UNIVERSITY(NATURAL SCIENCES EDITION)*, vol.31, no.3, pp.316-320, 2007.

[13] Maarouf M Y, Chung S M, XML Integrated Environment for Service-Oriented Data Management. Vol.2, pp.361-368, 2008.

[14] QIU Peng,CHEN Jian-hui,CHANG Qing,Application of XML Technology in Data Exchange Based on Internet, *INSTRUMRNTATION TECHNOLOGY,* no.12, pp.57-58, 2007

[15] HE Guo-hui,QING Yin-bo,Data exchange system design based on XML,COMPUTER ENGINEERING AND DESIGN, vol.28, no.3, pp.583-587, 2007.

[16] LI Zong-hua,ZHANG Lei,Lightweight Heterogeneous Data Integration Method Based on XML Schema,*Computer and Modernization*, no.11, pp.93-98, 2005.

[17] GU Fang-zhou, SHEN Bo.Application study on JSON data exchange format in integration of Heterogeneous System, *Railway Computer Application*, vol.21, no.2, pp.1-4, 2012.

[18] ZHANG Hu-yin,QU Qian-song,HU Rui-yun.Data exchange model based on JSON, *Computer Engineering and Design*, no.12, pp.3380-3384, 2015.

[19] DING Bo, CHAO Ai-nong, Research on AJAX development based on Struts2 framework, *COMPUTER ENGINEERING AND DESIGN*, vol.30, no.16, pp.3910-3913, 2009.

[20] GAO Jing, DUAN Hui-chuan, Research on data transmission efficiency of JSON, *Computer Engineering and Design*, vol.32, no.7, pp.2267-2270, 2011.

[21] YU Jinliang, ZHU Zhixiang, LIANG Xiaojiang, A Data Exchange System Based on Sqoop, *Internet Technology*, 2016, 6(3):35-37.

[22] XU Canfei, FANG Fang.Design and Implementation of Data Exchange Platform Based on ODI Enterprise Information Management, *Practical Electronics*, no.(3-5), pp.36-37, 2016.