A Domestic Database Replacement of Information Systems in Enterprise

Huichao Liang*, Yuan Liu, Di Wang, Man Guo, Lin Mei and Haibin Zhao

{*Corresponding author: lianghuichao1993@163.com}

{lovell.liu@gmail.com, wangdi199022@126.com, 13326324@qq.com, haibinzhao@126.com}
State Grid Henan, Information & Telecommunication Company Data Center, Zhengzhou, Henan, China

Abstract. With the deep application of information technology, a variety of data are steady flowing into the enterprise database, security issue becomes more and more important. In order to avoid the huge risks in data security, follow the independent and controllable strategy of information infrastructure, and promote high-quality digital development, State Grid Henan electric power company has launched a pilot application of domestic databases, relying on automated and visual tools. During the replacement process, we proposed four core steps, including database selection, compatibility evaluation, data migration, and data validation. We also summarize typical problems and handling cases to promote applications better adapt to the new database. By replacing foreign databases with domestic databases, the security and performance of information systems have been greatly improved. In addition, the handling of typical problems in this paper also has a reference significance for the replacement of other domestic databases.

Keywords: Domestic Database, Data Migration, Enterprise Informationization, Visual Tool

1. Introduction

With the rapid development of information applications, how to help enterprises protect the security and stability of massive data has become an important research topic. Many information systems in enterprise are based on Oracle database environment [1], whose core technology always enslaved to foreign countries. In recent years, the database technology developed by Chinese has gradually gained the favor of enterprises in China due to its independent intellectual property rights, secure technology, reliable after-sales service, and affordable prices. To carry out a pilot application of domestic database in enterprise, the following challenges need to be addressed:

(1)There are many types of domestically produced databases[2], which are in different technical routes, so it's difficult to select the appropriate domestically produced database.

(2)Domestic databases differ in structure from traditional databases[3], so it is inevitable to encounter syntax incompatibility, application driver incompatibility,

ICSETPSD 2023, November 17-18, Coimbatore, India

DOI 10.4108/eai.17-11-2023.2342769

function failure[4] and so on.

(3)After completing data migration[5], we may face problems such as inconsistent data validation and unstable system running[6].

2. Main Approach

In order to efficiently realize database replacement, we propose a solution, which includes four steps:domestic database selection, compatibility evaluation, data migration, and data validation. Throughout the process, we make full use of visual tools [7] to improve the accuracy and efficiency, and continually optimize and adjust parameters[8] to solve compatibility problems.

2.1 Select the Appropriate Domestically Produced Database

Based on information system research and technical demonstration, we summarize the core factors that affect the selection of domestic databases:

(1)Database performance[9] : verify whether there are any shortcomings or defects in the overall performance of the domestic database, and whether the database can meet the high reliability requirements in terms of database infrastructure, SQL engine, storage engine, etc.

(2)Heterogeneous compatibility: verify the compatibility of the domestic database with the original environment, and whether it can provide comprehensive compatibility support in terms of SQL syntax, data dictionary,performance view, data type, built-in function, application interface, etc.

(3)Security controllability: judge whether the domestic database software is autonomous and controllable, whether it is designed by Chinese with complete independent intellectual property rights, whether it can grasp key technology,and whether it can guarantee the security and reliability of database access and data storage[10].

2.2 Intelligent Compatibility Evaluation

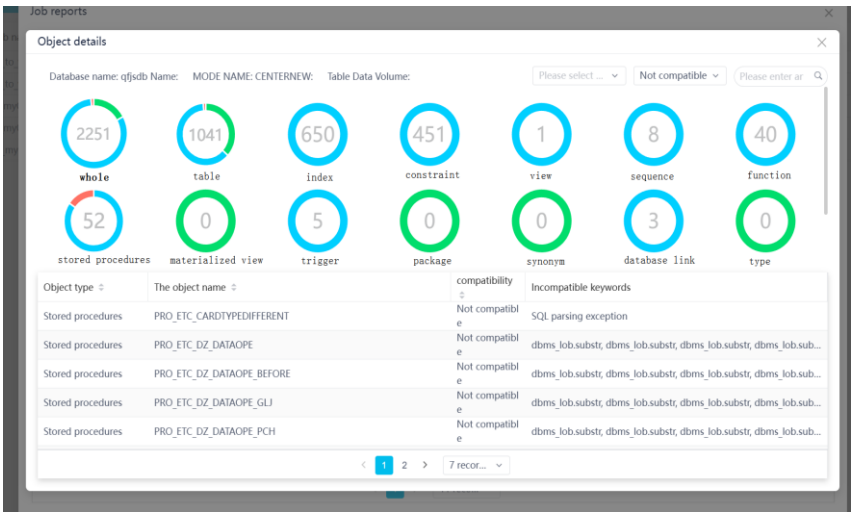

Fig.1 Intelligent compatibility evaluation by visual tools

Before the formal replacement, we must understand what incompatibilities exist. To improve efficiency, we use visual tools, which provide the function of intelligent compatibility evaluation before data migration, to automatically analyze the compatibility of migrated objects and help us positioning the cause of the incompatibility.

2.3 Data Migration

In the data migration phase, we design different module programs such as online full data migration, real-time data synchronization and incremental data subscription in different job scenarios, to monitor the every part of lifecycle in data migration.

After the accomplishment of migration, a migration report will be automatically generated. Based on the migration report, the migration results will be clearly displayed, and specific problem analysis and handling will be carried out for unsuccessful objects.

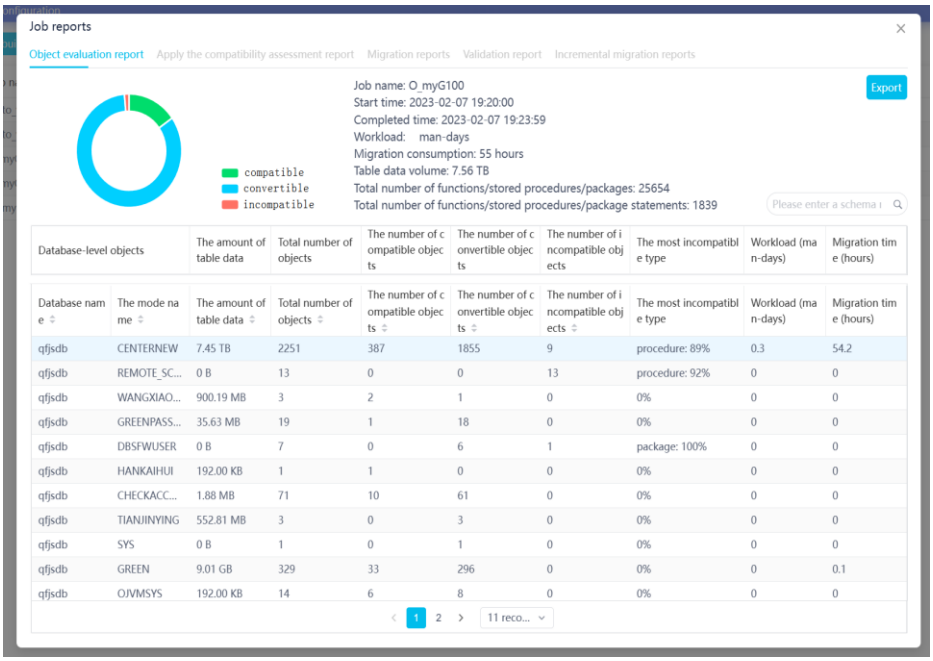

Fig.2 Automatically-generated migration report

2.4 Data Verification

After the data migration, it is necessary to compare the data before the migration and the data after the migration. Through visual tools combining multiple data validation methods, which can showcase differential data and provide fast repair capability, we visually verify the integrity of the migrated data,in that way, data accuracy will be comprehensively and efficiently ensured.

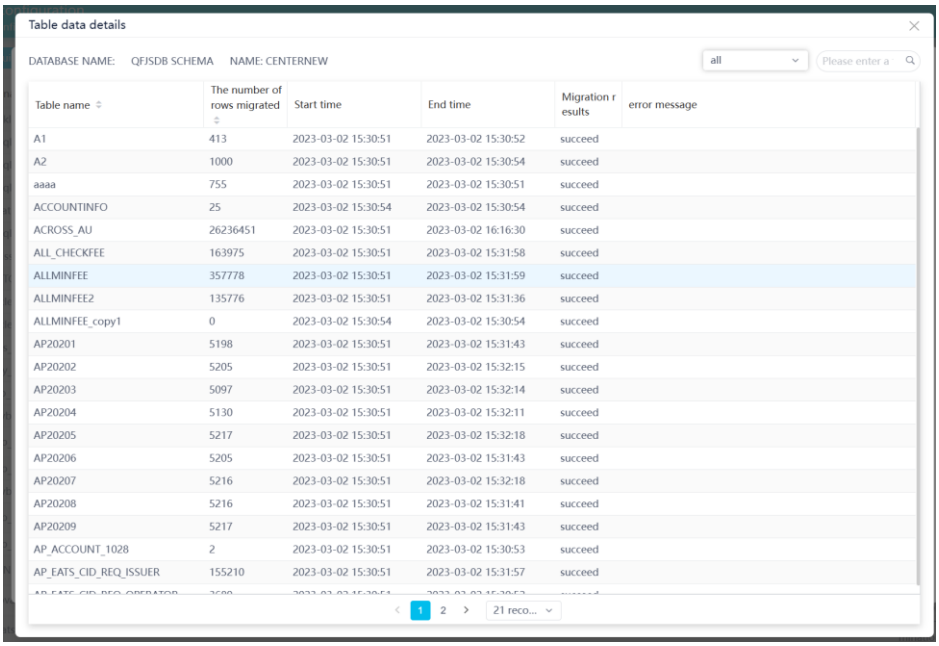

Fig.3 Migrated data verification

3. Applications Adjustments

3.1 Usual Problems

After completing the database migration, the application adaptation work must be carried out. We have summarized the main problems encountered during the adaptation process as follows:

(1)Incompatibility between the version of the application development platform and the database. At different times, the application development platform varies, and the JDK version varies. Therefore, the application needs to be adjusted to specific databases to achieve consistency between the program and data.

(2)The connection driver and interface specification are inapplicable to the new database. Since the application program used Oracle Database before, after replacing the database with a domestic database, the interfaces and drivers also should be replaced correspondingly.

(3)Functional module and business process of the system are abnormal. There are some differences in SQL statement between the domestic database and Oracle Database. The domestic database is more rigorous in logic, especially in terms of query results sorting, field comparison, field type statistics, etc. Therefore, it is necessary for software architect and developer to further optimize the system program in combination with the new database language.

3.2 Typical Problem Handling Case

To help readers grasp how to handle issues between applications and databases intuitively, we present a detailed problem handing case for a typical error.

After the completion of data migration, an error is reported when the application is started (Fig 4):

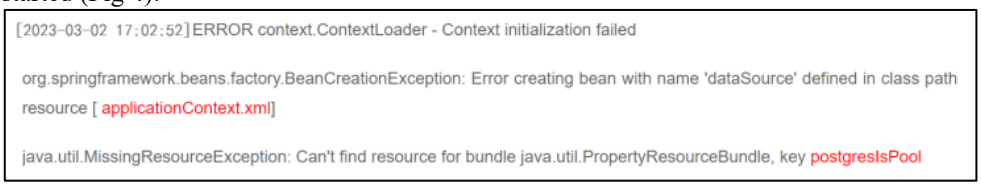

Fig.4 Error message

According to the log prompt, there are no key and value pairs related to postgresIsPool referenced by the data source dataSource in applicationContext.xml. We view the configuration of applicationContext.xml as follows:

```
<bean id="propertyConfigurer"
    class="org.springframework.beans.factory.config.PropertyPlaceholderConfigu
rer">
        <property name="locations">
            <list>
    <value>classpath:/parameter.properties</value>
            <value>classpath:/stat.properties</value>
                <!--value>classpath:/jdbc.properties</value>-->
                <value>classpath:/cams.properties</value>
            <value>classpath:/sso.properties</value>
            </list>
        </property>
    <property name="ignoreUnresolvablePlaceholders" value="true" />
    </bean>
    <bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource"
        destroy-method="close">
        <property name="driverClassName">
            <value>${jdbc.driverClassName}</value>
        </property>
        <property name="url">
```

```
            <value>${jdbc.url}</value>
        </property>
        <property name="username">
            <value>${jdbc.username}</value>
        </property>
        <property name="password">
            <value>${jdbc.password}</value>
        </property>
    </bean>
```

The applicationContext.xml references the parameter.properties parameter file which in the same directory. By Further examination of the parameter.properties parameter file, it shows that there are only Oracle related configurations like the following, but no postgreSQL configuration:

```
DB = POSTGRESQL
POSTGRESQLDRIVER = ORG.POSTGRESQL.DRIVER
POSTGRESQLURL
 JDBC:POSTGRESQL://X.X.X.X:5432/ORCL
POSTGRESQLUSER = XXXXXX
POSTGRESQLPASSWORD = XXXXXXXXX
POSTGRESQLISPOOL = C3P0
POSTGRESQLPOOLNAME = JAVA:COMP/ENV/JDBC/FSM
POSTGRESQLTYPE = POSTGRESQL
```

According to the above Oracle data source parameter configuration, we edit the postgreSQL configuration as follows:

```
DB = POSTGRESQL
POSTGRESQLDRIVER = ORG.POSTGRESQL.DRIVER
POSTGRESQLURL = JDBC:POSTGRESQL://X.X.X.X:5432/ORCL
POSTGRESQLUSER = XXXXXX
POSTGRESQLPASSWORD = XXXXXXXXX
POSTGRESQLISPOOL = C3P0
POSTGRESQLPOOLNAME = JAVA:COMP/ENV/JDBC/FSM
POSTGRESQLTYPE = POSTGRESQL
```

After modifying the configuration, the Java application connects to the postgreSQL database successfully.

As this typical problem handing case demonstrated, we can realize that the configuration of domestic databases is not complicated, and it won't cost database administrators much trouble. By analyzing logs and changing simple configuration files in domestic databases, most adaptation issues can be quickly resolved.

4. Effects

4.1 Save Database Software Costs

Domestic databases have a good cost-effectiveness ratio, and their functionality and performance can meet the needs of the information system. Compared to foreign databases such as Oracle, domestic databases are more affordable. By replacing a foreign database with a domestic database, the license fee for each set of database software can be saved by 100,000 yuan, the maintenance cost for each set of database software can be saved by 20,000 yuan per year.

4.2 Improve Data Security

Domestic databases are independently developed by Chinese, mastering the core technology, wholly owning the intellectual property right. Enterprises do not need to worry about information security when using domestic databases.

4.3 User-friendly Support

Manufacturers of domestic databases have local technical teams with multiple engineers, the technical support service from domestic database software vendors responds quickly, making domestic databases highly advantageous in the information industry.

4.4 Provide Open Functionality

Databases are basically important for information infrastructure. Domestic databases have extremely high openness and flexibility, they support customized development and functional expansion to meet the needs of different users, ensuring the stable running of information systems, bringing greater quality and efficiency.

5. Conclusions

This paper puts forward some solutions and working ideas in the pilot application process of domestic databases and it has great reference significance for the development of information technology in Chinese enterprises. In the process of data migration, many typical compatibility problems have been solved, laying the foundation for the further research of domestic information technology [11].

As domestic databases gradually occupy a place in enterprise informatization, more domestic technologies will be applied to the entire chain of enterprise informatization. Due to the success of this domestic database replacement pilot, we will continue to execute domestic replacement in more fields (such as basic software, application, desktop terminal and so on), promoting high-quality development of enterprise informatization and ensuring information security [12].

References

[1] Astrahan M M, Blasgen M W, Chamberlin D D, et al. System R: Relational Approach to Database Management. ACM Transactions on Database Systems, 1976, 1(2):97-137.

[2] Hong Z, Fan G, Yucai F, et al. DM Database Management System Web Service Research. Huazhong University of Science and Technology, 2007:4-8.

[3] Cheung T Y, Zhu Y. Recognizing different types of beta-cycles in a database scheme. Theoretical Computer Science, 1991, 81(2):295-304.

[4] Pourghebleh, Behrouz, Vahideh Hayyolalam. A comprehensive and systematic review of the load balancing mechanisms in the Internet of Things. Cluster Computing, 2020, 23: 641-661.

[5] Wu M, Jiao X, Li Y. Discussion of the Automatic Migration Scheme of Oracle Database. Proceedings of 2015 International Conference on Advances in Mechanical Engineering and Industrial Informatics(AMEII 2015), 2015:623-631.

[6] Diallo O, Rodrigues J J P C, Sene M, et al. Distributed Database Management Techniques for Wireless Sensor Networks. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(2):604-620.

[7] Mutale Kasonde, Simon Tembo. A Seamless Network Database Migration Tool for Insititutions in Zambia.International Journal of Advanced Computer Science and Applications (IJACSA), 2018, 9.

[8] TrilloMontero David, CosanoLucena Samuel, GonzalezRedondo Miguel, et al. Design and Development of a Relational Database Management System (RDBMS) with Open Source Tools for the Processing of Data Monitored in a Set of Photovoltaic (PV) Plants. Applied Sciences, 2023, 13(3).

[9] Taipalus Toni, Grahn Hilkka, Ghanbari Hadi. Error messages in relational database management systems: A comparison of effectiveness, usefulness, and user confidence. The Journal of Systems & Software, 2021, 181.

[10] Wu H, Cheng C. Research on Search Optimization Based on Oracle Database. Journal of Physics: Conference Series, 2020, 1648(4).

[11] Benenson M Z, Alekseeva E A. Building a database for complex industrial monitoring systems. Radio Industry (Russia), 2021, 31(1):65-73.

[12] Thuraisingham B. Security and privacy for multimedia database management systems. Multimedia Tools and Applications, 2007, 33(1):13-29.