

Architecture Design of Wide-area Distributed Real-time Database in Grid Dispatching Control System

WANG Zhihua¹, LIU Jin^{2,a}, SU Fan¹, SUN Mingyang^{2,b}, FAN Jing¹ and ZHUANG Weijin^{2,c}

¹State Grid Shanghai Electric Power Company, Shanghai, China

²China Electric Power Research Institute, Nanjing, China

^aliujin@epri.sgcc.com.cn, ^bsunmingyang@epri.sgcc.com.cn, ^czhuangweijin@epri.sgcc.com.cn

Keywords: grid dispatching control system; wide-area; distributed real-time database; data dictionary

Abstract. Power grid in future will develop into an extra-large interconnected grid with larger transmission capacity and more complex running characteristics. The trend of grid dispatching control system must be global wide-area and dynamic integrated necessarily. In this situation, it makes higher demand for real-time data to support grid supervisory and analysis. However, the existing real-time database with a centralized architecture commonly had the limits of scalability and performance, and it could hardly support a large number of concurrent data access. This paper proposed an architecture design of wide-area distributed real-time database (WDRDDB). WDRDDB divided all the real-time data into parts, and stored them on several servers. In addition, WDRDDB introduced a novel kind of data dictionary to index all data to provide a unified data access service for whole grid. Through experiments, we proved that WDRDDB had a better performance in real-time data processing than existing real-time database system.

Introduction

Grid dispatching control system is an important means of power grid coordinating and controlling [1,2,3], and it works based on grid real-time data which is the foundation of making grid analyses, evaluations and strategies. Real-time data is so important that almost all applications of dispatching control system are running upon it, such as SCADA, PAS, AVC, AGC et al. So, as the storage of real-time data, real-time database is one of the most critical modules in grid dispatching control system, and it is also the key technology to make a safe, steady and reliable grid [4,5].

Power grid in future is becoming an extra-large interconnected grid with larger transmission capacity and more complex running characteristics. The trend of grid dispatching control system would turn to be global wide-area and dynamic integrated. In this situation, it makes higher demand for real-time data to support grid supervisory and analysis applications [6].

However, the existing real-time database (RTDB) using a centralized architecture commonly had the limits of scalability and performance, and it could hardly support a large number of concurrent access. So, it is very necessary to research on the technology of distributed real-time database [7]. The distributed real-time database could be used to process real-time data, including measuring data, calculation result and temporary data. It is designed to improve the efficiency of concurrent data processing.

This paper proposed an architecture design of wide-area distributed real-time database (WDRDDB). WDRDDB divided all the real-time data into parts, and stored them on several servers. In addition, WDRDDB introduced a novel kind of data dictionary to index all data to provide a unified data access service for whole grid. Through experiments, we proved that WDRDDB had a better performance in real-time data processing than existing real-time database system.

Design Idea

WDRRTDB is designed following the concept of “physical distribution and logical integration” and is developed based on the existing real-time database system. It mainly includes two parts in details:

Physical Distribution. In existing grid dispatching control system, both model and measuring data are stored in one data record. This type of data structure is simple but not easy to extend. It’s hard to make horizontal scaling according to system requirements. Differently, in WDRRTDB, static data and dynamic data are separated based on the data property. That is to say, model-related and parameter-related static data are moving to one server, and measure-related dynamic data is saved on distributed data nodes. Those data nodes form a cluster to store measuring data which is divided into parts according to substation. In this way, WDRRTDB implements distributed storage for real-time data to improve the system horizontal scalability and the efficiency of massive concurrent data accessing.

Logical Integration. WDRRTDB provides data access service by standard APIs. All the applications in dispatching control system can read and write real-time data by these unified APIs. And all the data is fully integrated by data access service in different regions. Clients can access any real-time data in any region through WDRRTDB’s API if the application needs to. For clients, the distribution of physical data is insensible and they have no idea where the data is stored. In this way, all real-time data makes a logical integration which is available for whole grid.

As shown in Fig. 1, device-based model data and measuring-based data are separated and stored on different physical nodes. Measuring data is divided into parts according to substation and distributed among multiple data nodes by part. In the other hand, communication gateway, which is the transfer point of real-time data, forwards grid running data to SCADA systems. Then data processing programs process the data and save it on local real-time database. When a client sends a data request, data service dispatches the request to right data node according to data dictionary. The specific data node returns the response to client and completes one data request. WDRRTDB makes whole grid data shared and interconnected by this way.

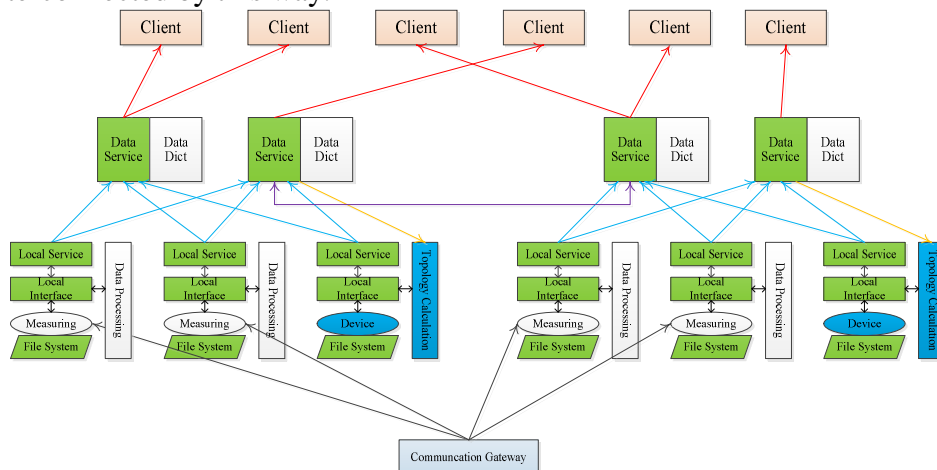


Fig. 1 Architecture of Wide-area Distributed Real-time Database

Implementation

WDRRTDB is a core content of future grid dispatching control system, is also the key module of “physical distribution and logical integration” architecture. The implementation mainly includes:

- 1) Distributed storage: separate different kinds of data, distribute real-time data;
- 2) Data dictionary: index distributed data by dictionary for data positioning and searching;
- 3) Unified data service: integrate whole grid data with unified APIs.

Existing real-time database system has made a good foundation for WDRRTDB. In order to implement WDRRTDB, we can upgrade some existing modules and develop core modules of distributed real-time database. Fig. 2 presents the software structure of WDRRTDB. The newly

developed modules include data access service, data dictionary and extended API for WAN connections.

Distributed Storage. There are two types of data stored in real-time database, model data and measuring data, which have totally different characteristics with each other. In existing real-time database system, the two types of data are stored together on the same server and it is becoming a bottleneck of system performance. So, according to the difference between different data, WDRRTDB clears up the mixed data and divides them into parts based on data attributes, and then distributes the data into a cluster. In this way, it improves concurrent processing ability and data access efficiency.

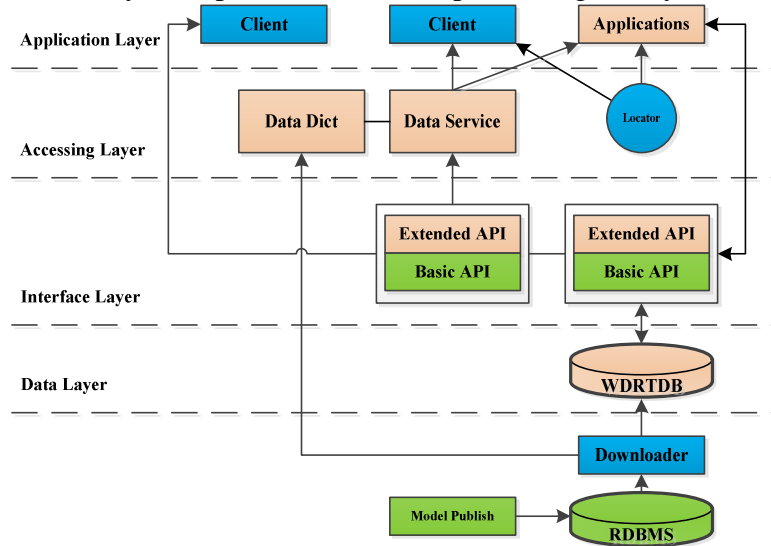


Fig. 2 Software Structure of WDRRTDB

Grid model data in dispatching control system belongs to static data. It includes device information such as substations, breakers and disconnectors et al. The model data hardly changes while system running. Only when grid model is maintained does model data update correspondingly. However, measuring data is changing all the time. After SCADA deals with grid latest running data, the data is saved into real-time database. Based on the different changing characteristics of data, static data and dynamic data should be separated and stored on different servers for better performance.

The mount of real-time measuring data is so huge that it could be a bottleneck of data access if putting all the measuring data on one server. Distributed storage is a right direction for performance improvement in this situation. As shown in Fig. 3, all the measuring data is organized in the form of group based on device's substation, and these groups are divided into several subsets. Each subset of data is saved on one server. These servers form a data cluster to provide data access service for clients and applications in whole grid.

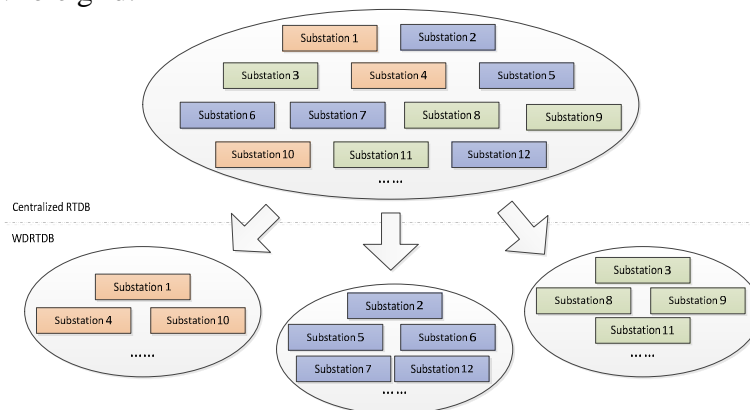


Fig. 3 Distributed Storage of Measuring Data

Data Dictionary. As the rule-maker when data service distributes access request, data dictionary plays a very important role in distributed real-time database system.

Data dictionary is designed for data indexing in WDRITDB system. It is organized in the form of items which record property information of original data resource, including name, location and owner. All the items are merged into a dictionary to help data service search and locate the data resource.

In existing dispatching control system, data resource is limited to dispatching region. Data from different regions is unavailable for each other. The limitation has reduced data usage a lot in whole grid. WDRITDB gets a great improvement by introducing data dictionary to index data of whole grid. Data dictionary is deployed with data service in all dispatching control systems, and applications can access data by using its unified APIs. As shown in Fig. 4, a typical data dictionary working process is commonly like this: an application sends a data request to data service with a set of IDs; data service looks up those IDs in data dictionary and groups IDs according to data position; Data service distributes sub data requests to WDRITDB system. In this way, application can access all the data within whole grid.

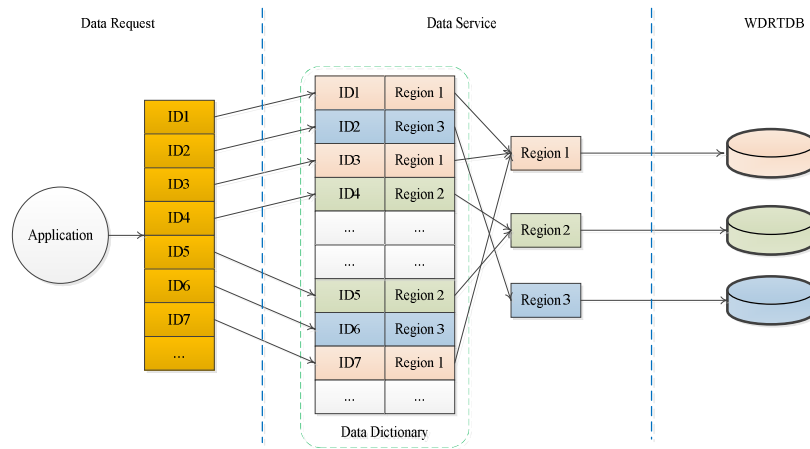


Fig. 4 Data Dictionary Indexing Process

Data Service. As the transfer point of data accessing, data service uniformly deals with data requests from all over the grid. It communicates with clients directly to provide data access service. Data service can parse data request according to data dictionary and distributes sub requests accordingly to right data servers, and finally merges all the sub results into the final response as shown in Fig. 5.

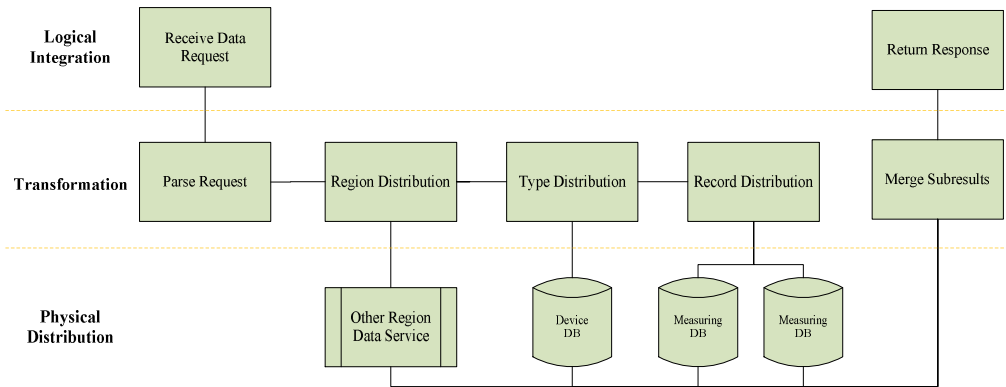


Fig. 5 Data Service Working Process

There are three stages in data request parsing and distribution as follows:

- 1) **Region Distribution.** If a data item requested by clients is not located in local region, local data service would forward this part of request to right data service in outside region based on data dictionary.
- 2) **Type Distribution.** As static data and dynamic data are separated on different servers, the request need to be further parsed and distributed according to data type.

- 3) Record Distribution, also named as substation distribution. Data service should access specific measuring data from specific data node because measuring data is distributed in a cluster grouped by substation.

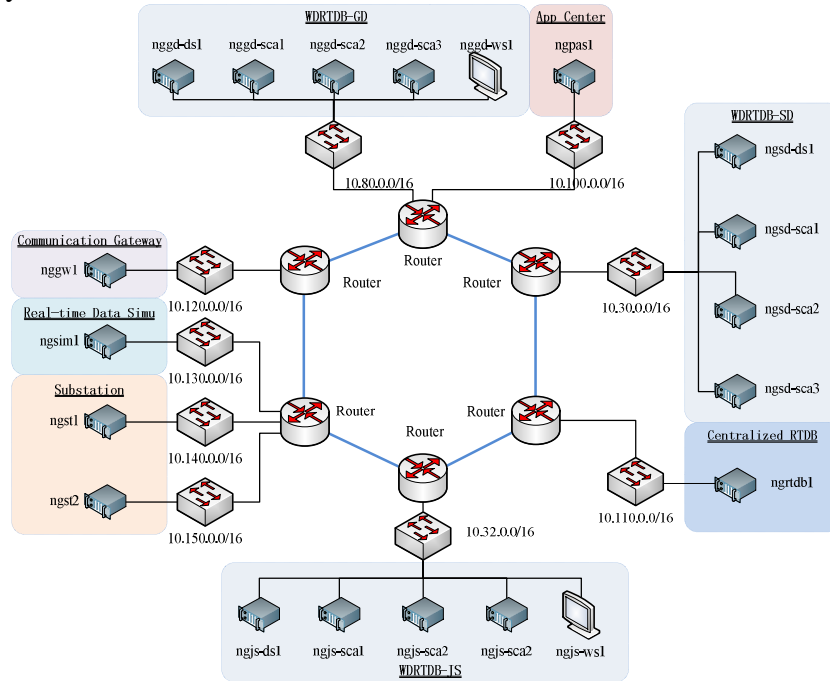


Fig. 6 Network Topology of Experimental Environment

Experiment Validation.

Experimental environment. Multiple different systems, including three WDRTDB systems and one centralized RTDB system, were set up and connected with each in WAN as shown in Fig. 6. Each WDRTDB system consisted of one data service server and three data node servers. The network bandwidth of all links in our experiment was set at 10000Mbps.

Table 1 Model Scale of Grid

No.	Device	Number
1	Substation	1444
2	Bus bar	4438
3	Breaker	15932
4	Disconnecter	43453
5	AC line segment	3711
6	Power transformer	882
7	Generating unit	606
8	Energy consumer	3189
9	Compensator	1021

Table 1 describes the information about grid model scale used in experiments. And Table 2 lists items about computing scale of state estimation.

Table 2 Computing Scale of State Estimation

No.	Item	Number
1	Substation	1444
2	Computing node	2890
3	Valid measuring point	28321

Experimental method. Testing with traditional centralized RTDB and WDRTDB respectively, we evaluated system performance by following aspects: SCADA data reading time, state estimation computing time, writing time for computing result to device tables and writing time for computing result to local tables. In this method, we could have a general understanding of WDRTDB performance from the perspective of applications.

Table 3 Real-time database test results

Item	Centralized RTDB (second)	WDRTDB (second)
Whole grid SCADA data reading time	0.40	0.18
State estimation computing time	1.10	1.18
writing time for computing result to device tables	1.51	1.07
writing time for computing result to local tables	0.11	0.43
Total time	3.01	2.86

Results analysis. The test results are shown in Table 3. We could find that WDRTDB reduced 55% time when reading whole grid SCADA data, and also saved about 29% time while writing device tables than existing RTDB. The reason is that WDRTDB processes data requests in parallel with a cluster while RTDB does serial processing on one server. When processing massive data, WDRTDB would have a better performance. However, there was no obvious advantage when writing data into local tables, because both WDRTDB and RTDB only made serial processing for local tables, and WDRTDB perhaps had more cost for distribution management. In conclusion, WDRTDB could satisfy the practical requirements of state estimation in whole grid situation.

Conclusion

Real-time database is a core component of grid dispatching control system. Traditional centralized real-time database system could hardly keep up with the development of future grid. This paper proposed an architecture design of wide-area distributed real-time database based on the new trend of grid development, and discussed the implementation in detail, including data distributed storage method, data dictionary indexing mechanism and data service design. Through experiments we found that WDRTDB had higher efficiency and better reliability than existing real-time database system. It could be a solution for real-time data interaction and sharing between different levels of dispatching control systems to better enhance grid safety and stability .

References

- [1] Xin Yaozhong. Development trend of power system dispatching automation technique in 21st century[J]. Power System Technology, 2001, 25(12):1-10.
- [2] Yao Jianguo, Yang Shengchun, Gao Zonghe, Yang Zhihong. Development trend prospects of power dispatching automation system[J]. Automation of Electric Power Systems, 2007, 31(13):7-11.
- [3] YAO Jianguo, YAN Sheng, YANG Shengchun, et al. Practice and prospects of intelligent dispatch with Chinese characteristics[J]. Automation of Electric Power Systems, 2009, 33(17):16-20.
- [4] ZHANG Shenming, BU Fanqiang, YAO Jianguo, et al. Realtime database management system(DBMS) that conforms to IEC 61970standard[J].Automation of Electric Power Systems,2002,26(24):26-30.

- [5] LU Xingquan. Real time database technique in energy management system[J]. Automation of Electric Power Systems, 2003, 27(6): 21-24.
- [6] Yao Jianguo, Yang Shengchun, Chan Maohua. Reflections on operation supporting system architecture for future interconnected power grid[J]. Automation of Electric Power Systems, 2013, 37(21): 52-59.
- [7] Zhai Mingyu, Wang Jin, Wu Qingxi, Jin Jing, Wei Na. Architecture and key technologies of wide-area distributed real-time database system for power dispatching automation system[J]. Automation of Electric Power Systems, 2013, 37(2): 67-71.