

## Design of Electricity Market Contract Management System Based on Metadata

Yanmin Guo, Ping Shao, Chunbo Huang  
Power Automation Dept.  
China Electric Power Research Institute, Nanjing,  
China  
guoyanmin@epri.sgcc.com.cn

Ji Zhang  
NARI Technology CO. LTD.  
Nanjing, China  
zhangj@sgepri.sgcc.com.cn

**Abstract**—Electricity trade contracts are an important basis for market settlements in China. In order to implement centralized management and extended maintenance for all kinds of contract, contracts are described by public properties and private properties in this paper. The public properties are modeled as fixed relational data structure, and private properties are modeled as fine-grained atomic data, which described by metadata and can register online. Through assigning different private properties for different contract type, the contract can configure flexible. Based on this, Batch maintenance functions and data check functions are designed further.

**Keywords**- contract management; metadata; electricity market; batch maintenance; data check

### I. INTRODUCTION

Chinese electricity market consists of long-term bilateral transactions and short-term market transactions. Long-term bilateral transactions ensure a stable market supply and short-term market transactions optimize resources [1-2]. They must be recorded as contracts, which are the basis for the market settlement. Therefore, the contract is important link to connect transactions and settlement in electricity market [3].

Now State Grid Corp are building unified technical support system served for all regional markets. Contracts show great differences in different regional markets, which affected by market policies [4] and settlement rules [5-6]. The difference can be summarized in three features. The one is the diversity of contract. With the increasingly active of electricity market, the type of contracts is increasing along with the constantly development of transactions variety. The two is the difference of contract. The contract exhibit different characteristics in different regional markets due to different market environments [7]. The three is the expansion of contract. With the development of the electricity market, the property of the existing contract extended frequently.

The old contract management system is modeled oriented to database and use the fixed relational data structure [8]. When a new contract property get added on, or an existing property changes, the data structure modifications and code rewriting is necessary. This will result in plenty of code maintenance works and response time delay.

In this paper, the new contract management system based on metadata is designed, which solve the above problem. It uses the object-oriented design methods, the properties of contract are divided into two parts, public properties and

private properties. The public properties are modeled as fixed relational data structure, and the private properties are modeled as fine-grained atomic data, which are described by metadata and online expansion is supported. Through the association of contract types and private properties, contracts can be flexible configured and expanded timely. Base on this, batch operation and data validation are designed further.

### II. METADATA TECHNOLOGY

#### A. Contract Properties Analysis

Analysis of the properties of electricity transactions contracts, we can find that some properties are common for all contracts, such as contract id, contract name, contract code, contract type, begin time, end time etc. And the other properties are private for contracts, which are used specially for a kind of contract and not used for other contracts. For example, “approved price”, “debug price”, “plan check price” are need for annual bilateral contracts. “Transaction price”, “settlement price”, “settlement market members” are need for power generation quota trade contracts. “Power purchase price from grid”, “power purchase price from power plant”, “punish margin” is need for direct transaction contract between large consumer and power plant. These messages are different, and can’t be stored by the same relational database table.

In this paper, contract properties are divided into public properties and private properties which use different storage strategy in database. The public properties are stored in database by fixed relational data tables, and contract ID is the primary key. The private properties, Learn from the data warehouse design ideas, are stored in database for atomic data. The primary keys are contract ID and property ID. The private properties are described with rich metadata information. Based on these metadata, storage and retrieval of private properties are designed, HMI display and data check are implemented. The data structure for contract is shown as figure 1.

#### B. Metadata technology

Metadata is data describing the data and its environment which is used to record business description of data and can help program to use data [9]. Now we define private properties of contract as tags. The tags are described through metadata which record the messages about database store and HMI display. The program is designed according to metadata.

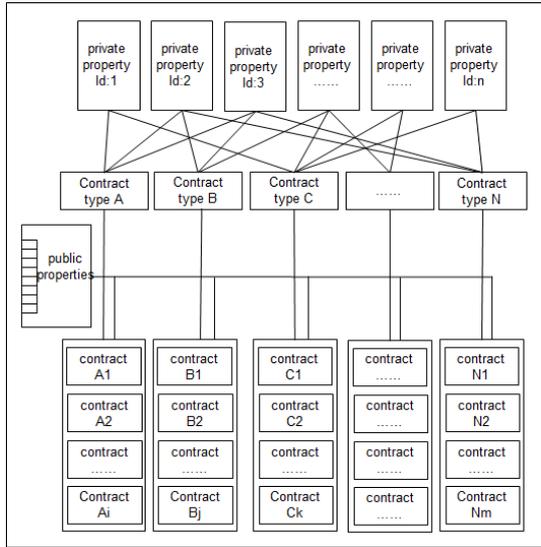


Figure.1 Data structure for contract

The metadata of tags can be summarized as several groups:

- i. The metadata describing base info, such as “tag ID”, “caption”, etc. ID is keyword of private property and automatically generated by the system. The format of ID is “XXYYYY”, where “XX” denotes value store table, “YYYY” denotes sequence number. Caption is used for HMI display.
- ii. The metadata describing data storage structure, such as “data type”, “data length”, “data precision”, etc. Through the metadata, table objects are defined for data store. There are number table, string table, time table, data table, blob table. Table objects can be added. The primary key for property value is contract ID union tag ID.
- iii. The metadata describing data reference relationship, such as “is references”, “reference object id”, “reference property id”, “reference display property id”, “reference menu”, etc. these info help program to implement HMI function. “Is references”=0, means that the private property have no reference info, interface appears as an ordinary input box. “Is references”=1, means that the private property refer to menu, interface appears as a drop-down box, in which options comes from menus definition. “Is references”=2, means that the private property refer to other tables. Options in drop-down box come from other tables.
- iv. The metadata describing display info, such as “data unit”, “display format”, “group info”, “display order”, etc. “Data unit”, matched with value, reflect the physical value, and display on the interface directly. “Display format”, describes numerical format requirements, by which data are processed before inserted into the database or after read from database. “Group info” and “display order” is used to optimize interface display when the contract has many properties.

- v. The metadata describing data validation rules, such as “max range”, “min range”, “is null” etc. these info help program to implement data check function.

TABLE I. THE LIST OF METADATA

Name	Description
Tag id	The primary key, sign a private property.
Caption	Chinese description for tag
Data type	Such as number, varchar2, date, blob, etc.
Data length	The total length of data
Data precision	Several retained after the decimal point
Is reference	Are there references info
Ref object id	If refer to other tables, gives table id.
Ref property id	If refer to other tables, give detail key field id of the table.
Ref display id	If refer to other tables, give detail field id for display.
Reference menu	If refer to menu, give menu code
Is format	Whether the data need to format for GUI
Display format	If format is need, give the format string
Data unit	reflect the physical meaning, display on GUI
Group info	Group for convenient view
Display order	Display order in the group
Max range	The maximum range
Min range	The minimum range
Is null	Whether the data is allowed to be null

### III. SYSTEM DESIGN

#### A. Database design

In the database, value of contract private property is stored and index through the primary key described as contract Id union tag ID. Five tables are designed to separately store numerical data, string data, date data, time data and blob data. For example, transmission power loss, as one of private properties, stored as a three field table which include “contract id”, “tag id”, “value”.

TABLE II. THE DATA STRUCTURE FOR TAG VALUE

Field	Caption
Contract id	The primary key contract public properties
Tag id	The primary key for contract private properties
value	Value for tag

The following diagram is the main data model of contract, describes data relation of references or dependencies with an arrow pointing, which reflects the business logic of the contract. The data model for contract is shown as figure 2.

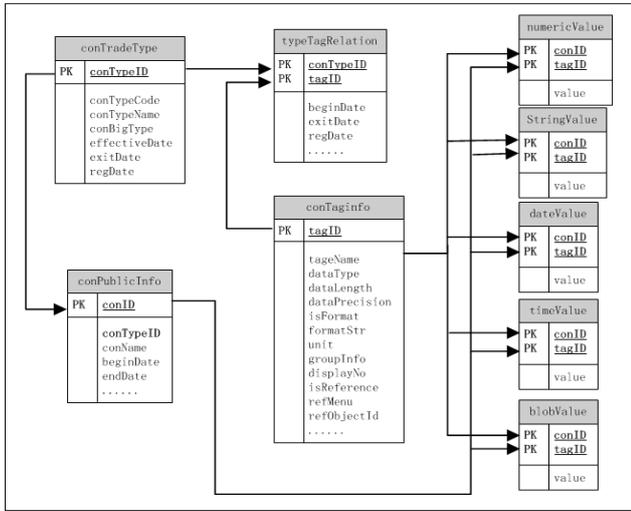


Figure.2 The data model for contract

### B. Function design

The contract management system is based on the Java2 platform, and uses the three-layer structure and design patterns based on model-view-controller (MVC) [10]. The technical structure is shown as figure 3.

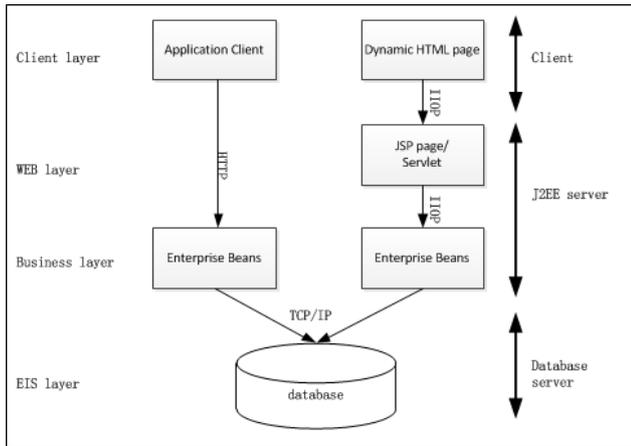


Figure.3 Technical structure for system

The system includes four core modules: Contract Input, Batch Contract, Data Check, Text Export. In addition, there is statistical analysis module for contract data, which is not described in this paper. The function structure is shown as figure 4.

#### 1) The function of contract input

The values of different properties for different contract are input and modified through HMI which is configured by program automatically rely on metadata definitions. There are five sub functions, such as contract types registration, contract properties registration, relationships set for type and properties, contract data input, contract monthly energy generation. Operating sequentially the five functions, we can

expand the contract type and contract properties, without modification of program and database.

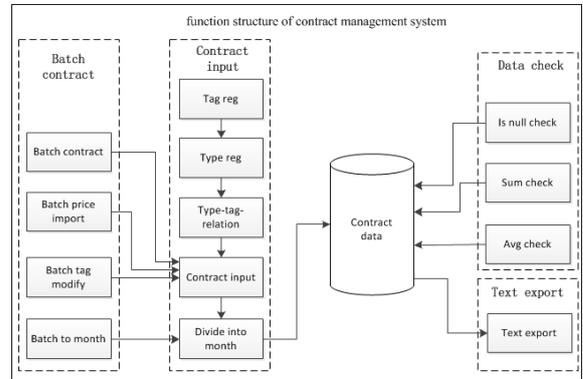


Figure.4 Function structure for system

#### a) contract types registration.

In this step, contract type can be added, deleted, modified. The public properties are described, such as caption, code, begin date, exit date, cycle, etc. There are many contract records under the contract type. The same contract type, the same contract properties.

#### b) contract properties registration.

In this step, the private properties are registered and metadata for them are defined. Detail description is shown in "Metadata Technology".

#### c) The relationships set for type and properties.

In this step, a contract type binds with many private properties. On the interface, we can select the contract type by drop-down box, check the properties by check box. When saved, the relationship between contract type and properties are stored. Rely on this relationship, contract show and store the value of related properties only. The GUI is shown as figure 5.

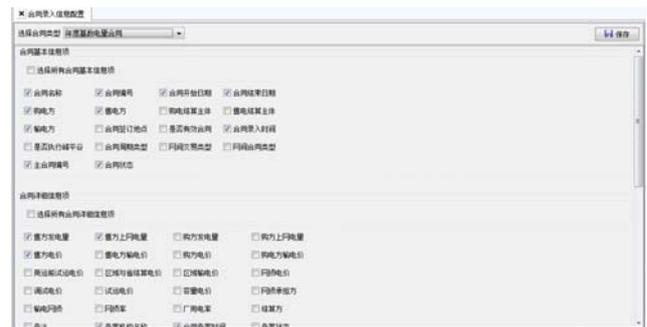


Figure.5 Relationships set for type and properties

#### d) Contract data input.

After the relationship build, the program automatically form input interface for different kinds of contracts. Each input items has different form of input items by reading metadata, such as text input box, drop-down box, time selection box and file upload box. According to metadata,

each input items can also have different data validation strategy, which ensure the data correctly.

*e) Contract monthly energy generation.*

Month is the shortest settlement cycle, so it is necessary to divide total transaction amount into monthly amount. The system provides several methods for split, such as “shared by months”, “shared by days”, “shared by Load forecast curve”, etc. Direct input is also supported. The monthly data are stored into fixed relation table.

*2) Batch contract*

Contract maintenance time is focused. For example, in the Jiangsu regional electricity market, there are more than two hundreds of annual bilateral contracts, which approved by the government's departments, need to be input into the system in a few days. In old system, operator must input contract data one by one through HMI. It's inefficient. Analysis of the same type of contract, we can find lots of info are same or similar.

- There are the same data, and can be input together. For example, in annual bilateral contracts for the same year, the contract data, such as power purchaser, power transmission participant, begin date, end data, backup date, etc., are the same.
- There are regular data. For example, the form of all contract names is sale participant name + year number (or other date number) + contract type.
- The same method is used to divide total transaction amount into monthly data for the contract with the same type and the same date.

For the above reasons, batch for contract maintenance is designed in the system. The functions include:

*a) Batch for contract initialization*

Select one contract type, select a group of the participants, and input a row of data for contract public properties, click initialize button. Then contracts are saved, whose name is a combination of participant names and fixed string data input.

*b) Batch for contract info import*

The system provides an interface, which can export contract private properties data to excel. In excel, the data quickly fill. Then click import button, select the file, the data are imported to the system. The rapid maintenance for contract is finished.

*c) Batch for monthly process*

Select contract and select property, then assign the method, the program will batch to divide total transaction amount into monthly data.

*3) Contract data check*

Since there are lots of contract records and each contract has lots of properties, it is easily happened that information is missing or incorrect. In the old system, manual inspection and verification is necessary. In this system, the function of contract check is designed. The program completes the basic inspection and verification, and gives the check result.

The steps of contract verification mainly include:

*a) verification rules set for single property.*

The verification rules for single property are set by the metadata, such as “is null”, “max range”, “min range”.

*b) verification rules set between properties.*

For the properties of contract, validation rules are configured for each other, including check horizontally between properties and check vertically between contracts.

If these is logic relation between properties, the data can be verified each other. For example, gross energy is greater than net energy generally. It is horizontal check. Vertical check is for sum or average of all contract value.

*c) Verification results query*

Finally, in the form of table list, check results are given for all contracts.

Contract verification results present to the operator through the interface, and assist the operator to check data integrity.

*4) Contract text export*

A contract template is prepared for a type of contract, in which contract property ids are inserted by the label {tag id}. When text export function is triggered, contract id is output to template. Union contract id and tag id, the program query the value from database, and replace the label. A contract text with real value is generated for print.

#### IV. CONCLUSIONS

This paper based on metadata, achieves the dynamic registration for properties of the contract and dynamic expand for contract structure, meet the actual expand needs of contract management. Further, the function of batch maintenance and the function of data check are proposed. These functions enhance the efficiency of the electricity market contract management, meet the expansion needs of contract, and improve the accuracy of contract data.

The system supports, but not limited to the following contract type: annual bilateral contracts between power plants and grid companies, power generation quota trade contract between power plants, electricity commit trade contract between power plants and grid companies, sale power contract to pumped storage units, direct trade contract between power plant and big consumer, inter regional energy transactions contracts.

#### REFERENCES

- [1] State electricity regulatory commission people's republic of china 2010 annual regulatory report of electricity trading and market, Beijing, August 2011.
- [2] SHANG Jincheng. Designing the electricity market system mode in China, part one: interconnected power grid market design [J]. Automation of Electric Power System, 2010, 34(8): 49-55.
- [3] CHENG Haihua, Yang Zhenglin. “The development of rule library and algorithm library for electricity market trading” [J], Automation of Electric Power System, vol.34, pp. 49-52, February 2010.
- [4] GENG Jian, GAO Zhonghe. “A preliminary Investigation on power market design considering social energy efficiency” [J], Automation of Electric Power System, vol.31, pp. 18-21, October 2007.
- [5] LONG Suyan, ZHANG Xian. “Research and application of flexible energy settlement methods based on component library”[J], Automation of Electric Power System, vol.37, pp. 89-94 July 2013.
- [6] CHENG Haihua, GENG Jian, “A settlement system design for

- provincial power generation enterprises based on algorithm library". Automation of Electric Power System, vol.35, pp. 99-101, December 2011.
- [7] ZHENG Yaxian, ZHANG Xian, "Energy Resource Optimization Model In Large-scale Based On Aggregate Node". IPEC2012,279-283.
- [8] WANG Xiaoyan, GU Jinwen. "Contract management system in power market"[J], Automation of Electric Power System, vol. 34, pp. 13-15, 2002.
- [9] YANG Hongbin, SONG Ming. "Research on architecture of metadata management platform"[J], Computer Systems Applications, vol. 11, pp. 17-20, 2007.
- [10] J. M. Steven, C. W. William, "Design Patterns in JavaTM (2nd Edition)", Addison-Wesley Professional, April 2006.