Advances in Engineering Research, volume 138

2nd International Conference on Machinery, Electronics and Control Simulation (MECS 2017)

A Novel SCD Validation Scheme Based on Information Classified Extraction and Semantic Strength Grading

Xu Gao[1)], Ke Wang[2, a)], Zhihan Jiang[1)], Yingxin Ma[1)], Yan Hu[2)] and Peng Peng[3)]

[1]*State Grid Jibei Electric Power Company, Beijing 100053, China.*
[2]*School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.*
[3]*Shanghai Yihao Automation Co., Ltd, Shanghai 201204, China.*

[a)]liuny0331@sjtu.edu.cn

Abstract. Substation Configuration Description (SCD), containing all the configuration information of a smart substation, is generally non-standard at present. SCD validation is essential to the normal operation of a smart substation. Typically, SCD is composed of some implicit information, secondary circuits for instance, which is hard to validate directly. In existing works, developing software is the most common method to validate implicit information, however, is quite complicated and inflexible. Against this, this paper proposes a novel scheme merely utilizes programs for information extraction and assigns validation to XML Schema due to its descriptive ability of rules. In addition, SCD correctness definitions are graded by their semantic strength in order to proceed the validation level by level. It is concluded that the novel scheme is not only good at describing validation rules, but also universal to all the application scenarios.

Keywords: SCD, information, XML

INTRODUCTION

With the enforcement of 'Smart Grid' strategy by State Grid, until the end of 2015, there are already over 3000 smart substations that have been commissioned and running with IEC 61850 communication standard. At present, China is leading in the massive deployment of smart substations [1-3]. Benefited from the communication protocol provided by IEC 61850, smart substations substitute numerous secondary wires with a so-called "three-layer-two-network" structure [4,5]. SCD encoded in XML (Extensible Markup Language) is the file describing complete substation detail, which contains substation, communication, IED, and Data type template sections. Admittedly, a standard SCD is essential to the normal operation, maintenance, and object-oriented modeling of smart substations in the context of massive deployment.

IEC 61850 recommends XML Schema to validate SCD, while merely for the syntactic structure [6]. In order to fully validate SCD, another schema language Schematron is combined with XML Schema, as Schematron is able to describe semantic constraints efficiently [7]. However, with Schematron added, the validation is still concerned on a single element or logical node in SCD. For a smart substation running with IEC 61850, wires between the relays and between the control house and the breakers are replaced with the communication infrastructure of SV and GOOSE messages. The wire reductions generate the virtual circuits which are implicit in SCD and difficult to validate directly. Therefore, the validation of virtual circuits is the core problem in SCD validation at present [8]. Aiming at validating virtual circuits in SCD, many validation schemes are proposed. Virtual circuits are determined by the virtual terminators on both ends, it is effective to check the existence of virtual circuits with their corresponding virtual terminators compared at once. As for the correctness of virtual circuits, it is validated through comparing "desc"(short for description) attribute with professional knowledge of power system [9]. Virtual circuits can also be divided into different types through which they will be validated in different methods [10]. However, all the validation schemes provided by existing works are realized through developing software. Complicated and inflexible, software is hard to describe and adapt to varying validation rules.

In general, programs are convenient to obtain all the information of SCD, meanwhile, XML Schema is good at describing validation rules. This paper proposes a novel validation scheme which combines information extraction by programming and validation through XML Schema.

XML SCHEMA

XML Schema is a schema language developed for XML by W3C. XML Schema specifies how to formally describe the elements in an XML document, meanwhile, it can be used to express a set of constraints to which an XML document must conform. An XML Schema file is conventionally written as an XML file, the namespace of which is "xs". Constraints are generally expressed using some combination of grammatical rules governing the order of elements, data types, elements, and attributes. Moreover, XML Schema can express specialized rules such as uniqueness realized by "xs: unique", and referential integrity constraints which are generally written as "xs: key" and "xs: keyref". As SCD is encoded in XML, IEC has provided a set of XML Schema files to describe the structure and elements in SCD according to IEC 61850. However, it is widely accepted by smart substation operators that more powerful constraints of SCD are needed in practical applications because SCD is not just a data file including many isolated elements.

On one hand, for traditional XML files, elements are treated equally if they have the same name. While in SCD, elements with the same name may have different structures and constraints according to one of their attributes. Therefore, a new method that can dynamically assign a type to an element based on an attribute is needed. Conditional type alternatives, a new feature designed in XML Schema 1.1, can achieve the above goal. It is written as “xs: alternative” and has two attributes: “type” and “test” [11]. An element in an XML instance document will be assigned to the “type” if its attribute values satisfy the “test”.

On the other hand, the configuration of the smart substation is organically integrated into SCD, connections between elements must conform to not only national and industrial standards but also power system knowledge. Assertions, also called co-constraints, in XML Schema 1.1 allow rules-based validation involving more than one element. Moreover, these rules are defined using XPath expressions, which is a very powerful query language for an XML document. Assertions are written as “xs: assert” in XML Schema 1.1 and have one attribute: “test” [11].

Benefited from the new features and improvements in XML Schema 1.1, SCD now can be validated not only in the aspect of syntactical structures but also flexible type definitions and some complicated constraints. As for the validation, it is conducted by comparing an XML file to the XML Schema document by tools like XMLSpy, Oxygen XML Editor or programs based on Xerces.

INFORMATION CLASSIFIED EXTRACTION

Although significantly improved in XML Schema 1.1, XML Schema is still limited in the validation of SCD. SCD is the whole configuration description of a smart substation, but it does not describe all the information equally. For the configuration of an element, SCD can explicitly describe it by constructing nodes and trees. However, connections between elements are hard to describe on the condition that SCD is in a structure of single element and node. These connections are implicit in SCD, for instance, the secondary circuits constructed by GOOSE and SV. For an IED, the output capabilities are configured as logical nodes, data objects, and data attributes, they are explicit and clear in SCD. While for the inputs of an IED, they are given in “ExtRef” with the names of opposite terminals. In order to get the details of the inputs, another IED must be checked by the information given in “ExtRef”. In a word, the description of a single connection cannot be interpreted without checking both terminals in two IED. Although assertions of XML Schema 1.1 are powerful in co-constraints, the test rules can be extremely complicated if connections are validated through assertions.

In order to validate the implicit information in SCD, an information extraction program is needed. Only if the information is explicit can XML Schema describe the corresponding rules. XML has two main parsers: SAX(Simple API for XML) and DOM(Document Object Model). SAX is an event-driven online algorithm for parsing XML documents, with an API developed by the XML-DEV mailing list, whereas the DOM operates on the document as a whole, i.e. building the full AST of XML document for the convenience of the user. SAX parsers operate on each piece of the XML document sequentially, issuing parsing events while making a single pass through the input stream. Due to the reason that SAX does not have to build a tree representation of the whole document, SAX is fit for parsing SCD which is a huge XML document [12]. In addition, common programming languages like JAVA, C++, and Python all have the interface of SAX, which means SCD is easy to be parsed.

There are diverse types of implicit information in SCD, and every type has its particular coupling manner, which calls for different decoupling programs. Therefore, implicit information has to be classified firstly. And then through parsing and analyzing SCD, the implicit information can be extracted. The final step is to write that information explicitly into a novel XML document and make it available for the validation through no matter a traditional syntactical constraint or a simple XML Schema assertion.

In generally, implicit information in SCD is classified firstly. And then it is extracted and explicitly written into a novel XML document by a program using SAX interface. Finally, a corresponding XML Schema file will be applied to validate the newly generated XML document, after which the validation of the specified implicit information in SCD is accomplished. It is particularly worth mentioning that the extraction program merely extracts, interprets and arranges the information. In another word, the validation of the XML file against a schema is technically separated from information extraction.

This article takes secondary circuits validation as an example of validating implicit information in SCD.

SEMANTIC STRENGTH GRADING

Generally, SCD correctness is a large concept including different levels of definitions. It is acknowledged that a high-level correctness definition is composed of several low-level correctness definitions. For instance, if an IED is considered to conform to Q/GDW 1396-2012 *Data Model of Protection Relay in Project Based on IEC61850* [13], then it certainly meets the requirements given by IEC 61850, due to the reason that IEC 61850 is the foundation of Q/GDW 1396-2012. In order to make different levels of SCD correctness definitions clear, it is necessary for low-level validations to be completed before high-level validations.

On the other hand, as the unique description file of smart substations, SCD is utilized by various users who may make different requirements on it. Manufacturers, for example, merely need to achieve the interconnections between elements and will not pay attention to the normalization of terminals names. State Grid, however, expects all SCD to conform to a large number of specifications given by standards such as IEC 61850 and Q/GDW 1396-2012. Therefore, different application scenarios demand different levels of validation. Based on this consideration, SCD correctness definitions are graded by their semantic strength and the validation of each level will be conducted separately.

Moreover, validation rules of the same element may vary with substation's properties, such as bus connection modes. In order to adapt to this diversity, different XML Schema documents aiming at different types of substations are created.

In order to make SCD validation logical and universal, this article presents a four-level SCD correctness definition as shown in Table 1:

TABLE 1. Four-level SCD Correctness Definition.

XML Schema documents	Validation objects	Standards	Examples
Level_1.xsd	Syntactical structures of SCD	*IEC 61850*	Element "SCL" must have a child element "Substation"
Level_2.xsd	IED configuration in SCD	Q/GDW 1396-2012 *Data Model of Protection Relay in Project Based on IEC61850*	LDevice configuration of 220kV line protection IED
Level_3.xsd	Primary system configuration and association between primary system and secondary system in SCD	*Technical Specification of System Configuration Description Files for Smart Substation*	Naming standard of IED instantiation
Level_4.xsd(Change along with bus connection modes)	Secondary circuits(virtual circuits) in SCD	Q/GDW 1161-2014 *Standardization Design Specification for Transmission Line Protection and Auxiliary Equipment* [14]	Secondary circuits configuration corresponding to line protection IED of double-bus mode

VALIDATION SCHEME

According to the four-level SCD correctness definition above, this article proposes a four-level SCD validation scheme, in which the following features are included:

(1) In order to confirm the validation level, every validation demand shall be estimated by its semantic strength firstly;

(2) A high-level validation must be premised on the successful validation of the previous level;

(3) Validation failures of any level shall with no exception lead to the modification of original SCD, after which the validation must start from level 1 again.

The four-level SCD validation scheme is shown in Fig.1:

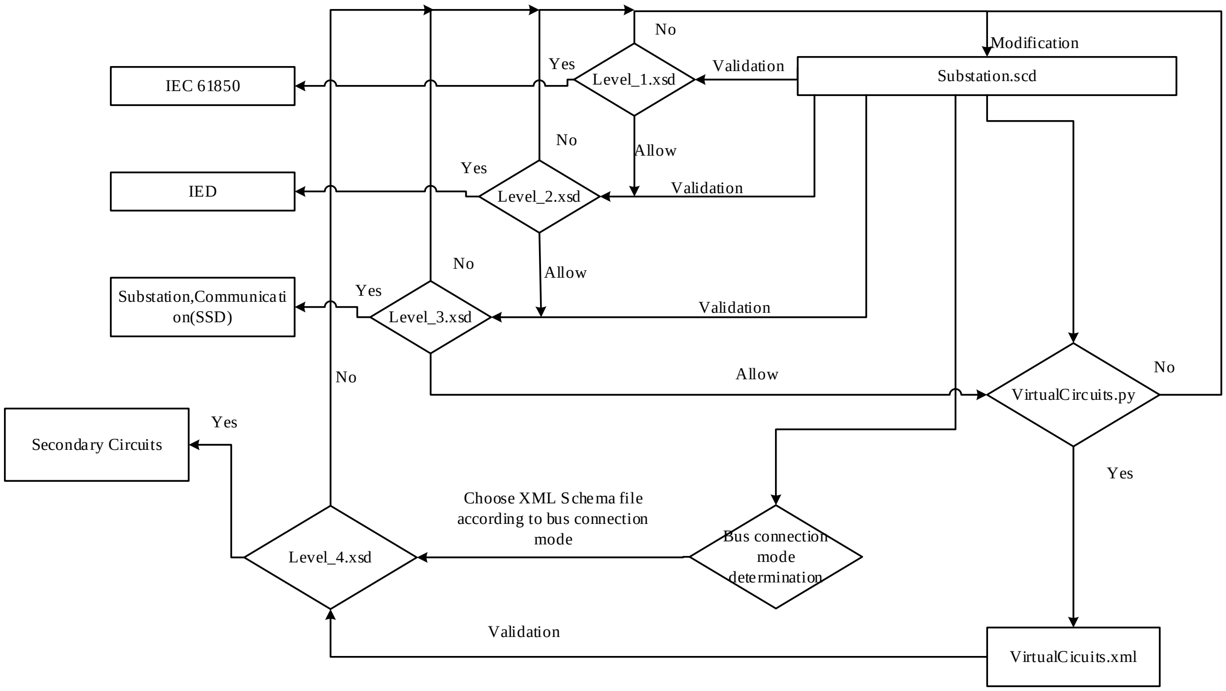


FIGURE 1. Four-level SCD Validation Scheme.

TESTING

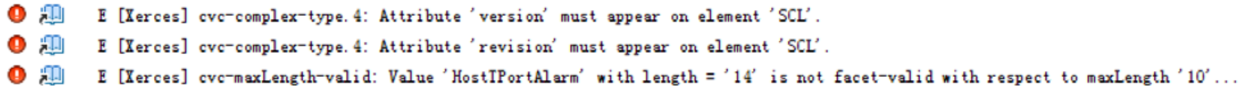
This article takes a 220kV HQ substation in a certain provincial grid as the testing example. Given that secondary circuits must be validated, a four-level SCD validation is to proceed.

(1) Level 1: IEC 61850 validation

i) XML document: Original substation.scd

ii) XML Schema document: SCL.xsd given by IEC 61850

iii) Part of validation results as shown in Fig.2:



```
E [Xerces] cvc-complex-type.4: Attribute 'version' must appear on element 'SCL'.
E [Xerces] cvc-complex-type.4: Attribute 'revision' must appear on element 'SCL'.
E [Xerces] cvc-maxLength-valid: Value 'HostIPortAlarm' with length = '14' is not facet-valid with respect to maxLength '10'...
```

FIGURE 2. Validation Results for Level 1.

Three errors mean:

(a) Element 'SCL' lacks attribute 'version';

(b) Element 'SCL' lacks attribute 'revision';

(c) The value length of element 'HostTPortAlarm' exceeds the maximum length.

(2) Level 2: IED configuration validation

i) XML document: Original substation.scd

ii) XML Schema document: Level_2.xsd according to Q/GDW 1396-2012 *Data Model of Protection Relay in Project Based on IEC61850*

Part of the standard logical nodes configuration of line protection IED is given in Table 2:

TABLE 2. Standard Logical Nodes Configuration of Line Protection IED.

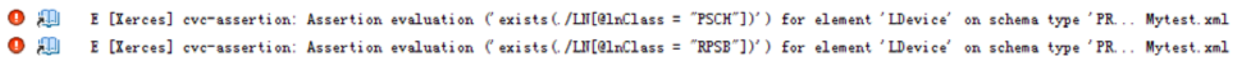
Logical Nodes	M/O	LDevice
LLN0	M	PROT
LPHD	M	
PDIF	O	
…	…	
PTOC	M	
…	…	

M means mandatory and O means optional

And part of the corresponding XML Schema document is:

```
<xs:complexType name="PROTType">
    <xs:sequence>
        <xs:element name="LN0" type="tLN0"/>
        <xs:element name="LN" maxOccurs="unbounded">
           …
        </xs:element>
     </xs:sequence>
     …
     <xs:assert test="exists(./LN0[@lnClass = &quot;LLN0&quot;])"/>
     <xs:assert test="exists(./LN[@lnClass = &quot;LPHD&quot;])"/>
     <xs:assert test="exists(./LN[@lnClass = &quot;PTOC&quot;])"/>
    …
 </xs:complexType>
<xs:simpleType name="tlnClass">
<xs:restriction base="xs:Name">
<xs:enumeration value="LLN0"/>
<xs:enumeration value="LPHD"/>
<xs:enumeration value="PDIF"/>
<xs:enumeration value="PTOC"/>
…
</xs:restriction>
</xs:simpleType>
```

iii) Part of validation results as shown in Fig.3:



```
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./LN[@lnClass = "PTOC"])') for element 'LDevice' on schema type 'PR... Mytest.xml
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./LN[@lnClass = "PSCH"])') for element 'LDevice' on schema type 'PR... Mytest.xml
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./LN[@lnClass = "RPSB"])') for element 'LDevice' on schema type 'PR... Mytest.xml
```

FIGURE 3. Validation Results of Level 2.

Three errors mean: LDevice 'PROT lacks logical nodes: 'PTOC', 'PSCH', and 'RPSB'.

(3) Level 3: Primary system configuration and association between primary and secondary system

i) XML document: Original substation.scd

ii) XML Schema document: Level_3.xsd according to *Technical Specification of System Configuration Description Files for Smart Substation*

Part of the naming standard of IED instantiation is given in Table 3:

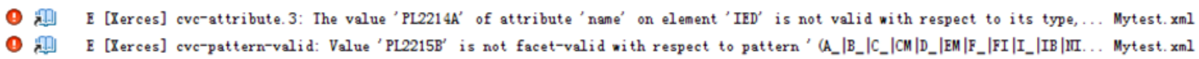
TABLE 3. Naming Standard of IED Instantiation.

1st Character 2nd Character	3rd Character	4th Character 5th Character	6th Character 7th Character	8th Character
IED Type	**Corresponding Primary Device Type**	**Voltage Level**	**Corresponding Primary Device Number**	**IED Number**
A_(auxiliary)	A(arrester)	00(for public)	The same as 'line'	A(First set)
B_(fault information)	B(breaker)	04(380V)	The same as 'line'	B(Second set)
…	…	10(10kV)	…	
P_(protect)	L(line)	…	500kV and above: last two characters of the corresponding switch; 220kV and below: bay sequence	…
PS(short-lead)	…	22(220kV)	…	
…		…		

And the corresponding XML Schema document is:

```
<xs:simpleType name="tIEDName">
      <xs:restriction base="xs:string">
            <xs:patternvalue="(A_|B_|C_|CM|D_|EM|F_|FI|I_|IB|NI|IP|L_|M_|MC|MV|MN|MG|MB|MI|P_|PN|PV|PS|
PC|PP|RF|RN|RM|S_|SP|SW|SC|T_)[A-Z](00|04|10|66|35|11|22|33|50|75|1k)\d{2}([A-D]|X)"/>
      </xs:restriction>
 </xs:simpleType>
```

iii) Part of validation results as shown in Fig.4:



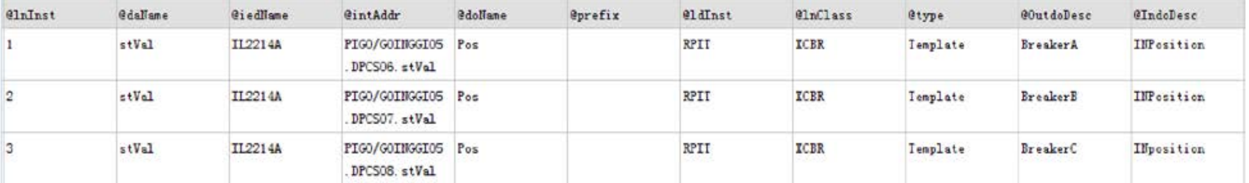
```
E [Xerces] cvc-pattern-valid: Value 'PL2214A' is not facet-valid with respect to pattern '(A_|B_|C_|CM|D_|EM|F_|FI|I_|IB|NI... Mytest.xml
E [Xerces] cvc-attribute.3: The value 'PL2214A' of attribute 'name' on element 'IED' is not valid with respect to its type,... Mytest.xml
E [Xerces] cvc-pattern-valid: Value 'PL2215B' is not facet-valid with respect to pattern '(A_|B_|C_|CM|D_|EM|F_|FI|I_|IB|NI... Mytest.xml
```

FIGURE 4. Validation Results of Level 3.

The errors mean: 'PL2214A' and 'PL2215B' are not valid IED names.

(4) Level 4: Secondary circuits

i) XML document: VirtualCircuits.xml extracted from original substation.scd, part of it is shown in Fig.5:



@lnInst	@daName	@iedName	@intAddr	@doName	@prefix	@ldInst	@lnClass	@type	@OutdoDesc	@IndoDesc
1	stVal	IL2214A	PIGO/GOINGGIO5 .DPCS06.stVal	Pos		RPIT	XCBR	Template	BreakerA	INPosition
2	stVal	IL2214A	PIGO/GOINGGIO5 .DPCS07.stVal	Pos		RPIT	XCBR	Template	BreakerB	INPosition
3	stVal	IL2214A	PIGO/GOINGGIO5 .DPCS08.stVal	Pos		RPIT	XCBR	Template	BreakerC	INposition

FIGURE 5. Part of VirtualCircuits.xml.

ii) XML Schema document: Level_4.xsd according to Q/GDW 1161-2014 *Standardization Design Specification for Transmission Line Protection and Auxiliary Equipment*

Part of the standard is given in Table 4:

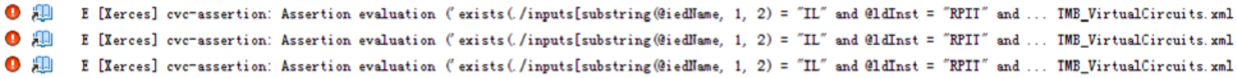
TABLE 4. Secondary Circuits Configuration Standard.

Signal Name	ldInst	lnClass	doName	dodesc
Breaker position of phase A	RPIT	XCBR	Pos	contains('A')
Breaker position of phase B	RPIT	XCBR	Pos	contains('B')
Breaker position of phase C	RPIT	XCBR	Pos	contains('C')
…	…	…	…	…

And the corresponding XML Schema document is:

```
<xs:assert test="exists(./inputs[substring(@iedName, 1, 2) = &quot;IL&quot; and @ldInst = &quot;RPIT&quot;
and @lnClass = &quot;XCBR&quot; and @doName = &quot;Pos&quot; and contains(@OutdoDesc,
&quot;A&quot;)])">
 </xs:assert>
<xs:assert test="exists(./inputs[substring(@iedName, 1, 2) = &quot;IL&quot; and @ldInst = &quot;RPIT&quot;
and @lnClass = &quot;XCBR&quot; and @doName = &quot;Pos&quot; and contains(@OutdoDesc,
&quot;B&quot;)])">
</xs:assert>
 <xs:assert test="exists(./inputs[substring(@iedName, 1, 2) = &quot;IL&quot; and @ldInst = &quot;RPIT&quot;
and @lnClass = &quot;XCBR&quot; and @doName = &quot;Pos&quot; and contains(@OutdoDesc,
&quot;C&quot;)])">
 </xs:assert>
```

iii) Part of validation results as shown in Fig.6:



```
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./inputs[substring(@iedName, 1, 2) = "IL" and @ldInst = "RPIT" and ... IMB_VirtualCircuits.xml
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./inputs[substring(@iedName, 1, 2) = "IL" and @ldInst = "RPIT" and ... IMB_VirtualCircuits.xml
E [Xerces] cvc-assertion: Assertion evaluation ('exists(./inputs[substring(@iedName, 1, 2) = "IL" and @ldInst = "RPIT" and ... IMB_VirtualCircuits.xml
```

FIGURE 6. Validation Results of Level 4.

Three errors mean: Break positions of phase A, phase B, and phase C are missed for the input of a certain line protection IED.

CONCLUSION

Based on XML Schema 1.1, this paper presents a novel SCD validation scheme, which has the following two features and advantages:

(1) For the validation of implicit information in SCD, the novel scheme separates extraction and validation. Through programming with SAX interface, any implicit information can be extracted, absolutely independent of SCD structure. Meanwhile, XML Schema 1.1, flexible and adaptable to rules, is applied to conduct the validation.

(2) SCD correctness definitions are graded by their semantic strength and the corresponding validations are separated into different levels. The test suggests that grading makes SCD validation logical and universal.

REFERENCES

1. P. Zhang, X. Gao, Intelligent substation, Electrical Technology, vol. 30, no. 8, 2010, pp. 4-10.
2. X. Gao, Intelligent substation technology, China Electric Power Press, Beijing, 2012.
3. Q. Huang, S. Jing, J. Li, D. Cai, J. Wu, W. Zhen, Smart substation: state of the art and future development, IEEE Trans. Power Delivery, vol. 32, no. 2, 2017, pp. 1098-1105.
4. Q/GDW 383-2009, Intelligent substation technology guidelines, State Grid Corporation of China, Beijing, 2009.
5. Y. Sun, H. Liu, J. Jiang, Analysis on completeness of substation configuration description file control strategy for smart substation, Automation of Electric Power Systems, vol. 38, no. 16, 2014, pp. 105-109.
6. IEC 61850, Communication networks and systems in substations, 2009.
7. D. Mei, R. Fan, B. Zhou, Research on regular expressions and check of IEC 61850 model information, Power System Protection and Control, vol. 43, no. 3, 2015, pp. 131-136.
8. D. Hu, J. Wo, Virtual circuit system of smart substations based on IEC 61850, Automation of Electric Power Systems, vol. 34, no. 17, 2010, pp. 78-82.
9. H. Wu, H. Dou, Q. Xiang, Y. Ma, The SCD file check method based on design specification of smart substation, Power System and Clean Energy, vol. 31, no. 4, 2015, pp. 84-88.
10. W. Bao, X. Gao, Q. Tu, B. Shen, J. Song, The development of SCD management method based on correct checking, Electrical Technology, vol. 6, 2016, pp. 133-137.
11. Information on http://www.w3.org/TR/2012/REC-xmlschema11-1-20120405/
12. E. R. Harold, W. S. Means, XML in a nutshell, third ed., O'Reilly Media, Sebastopol, 2004.
13. Q/GDW 1396-2012, Data Model of Protection Relay in Project Based on IEC61850, State Grid Corporation of China, Beijing, 2012.

14. Q/GDW 1161-2014, Standardization Design Specification for Transmission Line Protection and Auxiliary Equipment, State Grid Corporation of China, Beijing, 2014.