COMMUNICATION METHODS (PROTOCOLS, FORMAT & LANGUAGE) FOR THE SUBSTATION AUTOMATION & CONTROL

By MUHAMMAD UZAIR (Project Report Of Course 586 b)

ABSTRACT

Communication plays a vital role in automation & control in a substation. We have to collect information from the utility and exchange that information with the Master station in order to execute the control & monitoring actions. There should be some rules that define method of communication i.e Protocol, format & language. Many open and proprietary protocols i.e standard exist in the industry e.g Modbus, DNP3.0, Spabus. In this report we will discuss these protocols architecture, functionalities, device addressing and functions performed by each layer. New emerging standards i.e UCA 2.0 & IEC 61850 which provides interoperability for different vendor's equipments has been discussed. Also for a particular utility which protocol should be used has been analyzed.

Introduction

An electrical substation is a subsidiary station where the generation, transmission and distribution system of the electricity exists. Substation Automation Functions (SAS) is normally required to perform the control and monitoring of the different equipments The structure of the SAS is classified in 3 levels for the efficient control and monitoring of the substation. These are Station, bay and process level. This division is done based on their location and functional operation. To perform SAS functions, some sort of communication is required between the equipments placed in these levels. As data is required for all sort of further processing and to perform certain actions. IED (Intelligent Electronic Devices) is one of main system element of the SAS. It is a micro processor based device that implements specific functions such as control, protection and monitoring of the equipments in the substation. IEDs receive data from process side equipments and based on the data can issue control commands, such as tripping circuit breakers etc.

There are a number of different ways a protocol can control transmission of messages. It can be a Master / slave, Token ring and multiple access systems. SCADA and DCS systems communicate with their Remote Terminal Unit (RTU), or outstation equipment using a protocol. Initially some devices such as instruments and protective relays allowed remote communication via local RS232 connection or via dial-up modem (PSTN) link without a protocol. Soon however these devices also supported a protocol. The data representations sent are not identified in any fashion other than by absolute addressing. Each protocol consists of two message sets. One set forms the master protocol, containing statements for master station initiation and the other set is the RTU protocol, containing statements an RTU can initiate. The SCADA protocol between master and RTU forms a model for IED-to-RTU communications.

Why We need standards?

The communication protocol allows two devices to communicate with each other which uses the same protocol otherwise error will occur. So the utility is always restricted to one vendor. With the arrival of open systems concept, it is desired that devices from one vendor be able to communicate with those of other vendors i.e. devices should interoperate . Using open protocol gives the following advantages.

- Open system connectivity
- Supplier independence

- Low engineering costs
- Data (Information) availability.

There are some disadvantages also i.e more overheads, thus requiring higher speed for the same efficiency can not be obtained with Open standard protocol. Some of the existing Communication methods i.e protocols & standards in industry has been discussed below.

Modbus

Modbus protocol is an application layer messaging protocol with half duplex transmission and it is positioned at level 7 of the OSI model. It provides client/server communication between devices . Modbus is used to communicate intelligent devices with sensors and instruments & to monitor field devices using Master's stations. It became the first widely accepted fieldbus standard. The Modbus protocol family was originally developed by Schneider Automation Inc, as an industrial network for their Modicon programmable controllers. The device which requests the information is called the Modbus Master and the devices supplying information are Modbus Slaves. In a standard Modbus network, there is one Master and up to 247 Slaves, each having a unique Slave Address from 1 to 247. The Master can also write information to the Slaves. Masters can address individual slaves, or can initiate a broadcast message to all slaves. Slaves return a response to all queries addressed to them individually, but do not respond to broadcast queries. Modbus is an open serial communication protocol, meaning that it's free for manufacturers to build into their equipment.

The protocol does not define a physical network layer. The ASCII and RTU protocol operate on RS 232(Single Master-Slave configuration, Standard 8 bit ASCII protocol, baud arte-19200 bps), RS 422 and RS 485 (Single Master-Multiple slaves, Multi drop Connection, 100 Kbps) physical networks. The data is sent as series of ones and zeroes called bits. Each bit is sent as a voltage. Modbus/TCP protocol operates on all physical network layers supporting TCP/IP.

In order to store the information, Slave device uses four different tables. Two tables store on/off discrete values i.e coils and two store numerical values i.e registers. In Modbus language, a coil is a discrete output value. The coils and registers each have a read-only and read-write table. Each Table has 9999 values. Each coil is 1 bit and data address between 0000 and 270E. Each register is 2 bytes and has data address between 0000 and 270E. Coil/Register numbers are location names since they do not appear in the actual messages and Data Addresses are used in the messages. For example, the first Holding Register, number 40001, has the Data Address 0000. When the master requests data, the first byte it sends is the Slave address. So in this way slave knows whether this data is for him or not. Broadcasting is done by identifier 0. Broadcasts are unconfirmed, there is no guarantee of message delivery. The second byte sent by the Master is the Function code. This number tells the slave which table to access and whether to read from or write to the table. CRC is added (two bytes) added to the end of every modbus message for error detection. **ref(8)**

Mod bus also uses a Map which is basically a list of each individual slave device which defines about data (eg. pressure or temp.), where the data is stored (tables & data addresses) and how the data is stored (data types). This Map can be fixed or can be programmed according to the requirement.

Modbus Message Framing

Modbus message framing is used to mark the beginning and ending point of a message allowing the receiver to determine which device is being addressed and when the message is completed.

Partial messages detection and errors flag are also checked. Message frames are used to place a message by sending device. Each word of this message is also placed in a data frame that appends a start, stop and parity bit. For ASCII mode, the word size is 7 bits, and is 8 for the RTU mode. Thus every 8 bits of an RTU message is actually 11 bits. The structure of the data frame depends on the transmission mode i.e ASCII or RTU

Device Address
Function Code
Query Code
Error Check

Query message from Master & Slave (ref.2)

Device address field of the message has two characters in ASCII and 8 binary bits in RTU mode for slave addressing. The slave responds by placing its own address in this field of its response so the master know which slave is sending data.
Modbus function code field frame will contain two characters in ASCII and 8 binary bits in RTU mode. It will tell the slave what kind of action to take.
Modbus Data Field is formed from a multiple of character bytes i.e a pair of ASCII characters in ASCII mode and a multiple of two hex digits in RTU mode in range 00H-FFH. The data field typically includes register addresses, count values and written data. With no error, the data field of a response from a slave will return the requested data otherwise an exception code is returned.
Modbus Error Checking is performed either by parity checking or by the use of Cyclical Redundancy Check in RTU or Longitudinal Redundancy Check in ASCII mode. If the number of the slaves is more than the required 247 values , then 2 byte address is used for the slave address. Two byte addressing extends the limit on the number of slaves in a network to 65535.

The original Modbus interface ran on RS 232, but Modbus implementations is also done on RS 485 as it allows longer distances, higher speeds and the possibility of a true multi-drop network. Modbus can also be used for wireless communications and TCP/IP networks. There are two transmission modes in Mod bus, ASCII & RTU Mode. The choice depends on the different parameters i.e baud rate, parity etc. as part of the device configuration.

Mod Bus ASCII protocol uses hexadecimal ASCII encoding of data and 8 bit error checksum. The message frames contains ':' character at the beginning and a carriage return at the end. The ASCII messaging is less efficient and less secure than the RTU messaging and therefore it should only be used where RTU cannot be used. The ASCII messaging is state-less. There is no need to open or close connections to a particular slave device or special error recovery procedures.

Mod bus RTU protocol uses binary encoding of data and a 16 bit CRC check for detection of transmission errors. The message frames are surrounded by a silent interval of at least 3.5 character transmission times before and after the transmission of the message. When using RTU protocol it is very important that messages are sent as continuous character stream without gaps. If there is a gap of more than 3.5 character times while receiving the message, a slave device will interpret this as end of frame and discard the bytes received. Both ASCII & RTU uses the serial communication.

Mod Bus TCP is a TCP/IP based. It covers the use of Modbus messaging in both Intranet and Internet networks. It uses binary encoding of data and TCP/IP's error detection mechanism for detection of errors. It is a connection oriented protocol. It allows concurrent connections to the same slave as well as to multiple slave devices. So for each query it requires an cknowledgement.

It is low cost and also minimum hardware required to support it. Modbus TCP/IP uses TCP/IP and Ethernet to carry the MODBUS messaging structure.

DNP 3.0

DNP3 protocol was developed by Harris, Distributed Automation Products in 1993. DNP3 is an open protocol. The four core documents that define DNP3 are: Data Link Layer Protocol Description, Transport Functions, Application Layer Protocol Description and Data Object Library . DNP 3.0 uses the EPA layer Architecture.

User Layer
Application Layer
Data Link Layer
Physical Layer
Communication Medium

EPA Layer Organization(ref.2)

User layer acts as an interface between the OSI layer and application program. DNP3 is, in standard networking terms, a layer 2 protocol. It provides multiplexing, data fragmentation, error checking, prioritization, and layer 2 addressing services for user data. It makes particularly heavy use of CRCs embedded in its data packets. Many modern applications can now carry DNP3 messages over TCP/IP.

Data type and data size compose an information element. Data types are Unsigned Integers (UI) ,Integers (I),Bit string (BS),Octet string (OS). For example, BS12 specifies Bitstring of 12 its.The bit position in a data size n is denoted in square brackets [p1...pn] where p1 and pn are the first and the the last bits of the field.

For Physical layer, the DNP is specified over a simple serial physical layer such as RS-232 or RS-485 using copper (1200 bps), fiber, radio or satellite. It can also be used over an Ethernet connection. DNP3.0 provides utility a serial language to speak to substations.

The DNP 3.0 data link layer specifies the frame format, services, responsibilities, and transmission procedures for the data link layer. It basically provides those services which are required by the physical layer. The data link layer also relates the data link layer to IEC 870-5-1 and IEC 870-5-2 standards. The primary difference is that DNP 3.0 uses the FT3 frame format for asynchronous transmission rather than synchronous.

The DNP 3.0 layer stack adds a pseudo-transport layer, for the efficient transmission of large sized application data and it is not used by the IEC60870-5. This layer segments application layer messages into multiple data link frames. For each frame, it inserts a single byte function code which tells that this data link frame is the first frame or the last frame of a message, or both .It also includes a rolling frame sequence number which increases with each frame and allows the receiving transport layer to detect how many frames have been dropped. It is not a true end-to-end transport layer. All confirmation and reliability is provided by the data link layer. It results in reduced layers and overhead, and retains a high level of data integrity.

The DNP 3.0 application layer describes the message format, services, and procedures for the application layer. Once messages are built, they are passed down to the pseudo transport layer where they are segmented and passed to the data link layer and then passed to the physical layer. The application layer of the DNP3.0 is object based so it allows a range of implementations. The different modes like Polled only, quiescent mode and mixture of modes can be used.

If the data to be transmitted is very large for a single application layer message, multiple application layer messages are built and transmitted one by one. Each of the message is an independent application layer message and it is called as a fragment . It can be a single-fragment or a multi-fragment message. From the Master's side, these fragments are called requests and from the slave side they are called responses to those requests. A Slave DNP3 station may also transmit a message without a request i.e an unsolicited response.

Each fragment begins with a header followed by one or more object header combinations. The application layer header contains an application control code and an application function code. The application control code contains information about the fragment if it is one of a multi-fragment message or confirmation is requested or it is unsolicited fragment and also has a rolling application layer sequence number. The application layer header function code also contains the purpose or requested operation of the message. It applies to all object headers and to all data within the message fragment. The DNP 3.0 data object library document describes the format of data presented within an application layer message.

PROFIBUS

A Process Field Bus is a standard for field bus communication in automation technology and it was promoted by BMBF (german department of education and research) in 1989. The communication is between a Master central unit and Slave units. The data exchange is cyclic; the Master unit reads the Slaves input data and writes the Slaves output data. The Baud Rates for the GE Profibus-DP interface are from 9.6 kbit/s to 12 kbit/s. RS 485 serial link is used, the maximum number of the Slaves connected to the Bus is 125. This protocol is implemented by layer 2 of the OSI reference model.

At start, PROFIBUS FMS (Field bus Message Specification) was made but it was bit complex in its protocols. So in 1993, the specification for the simpler and considerably faster protocol PROFIBUS DP (Decentralized Peripherals) was completed. It replaced FMS. PROFIBUS has become the world's most popular fieldbus in discrete manufacturing and process control. Industries using PROFIBUS range from critical petrochemical operations plants to food, drink & waste treatment plants.

PROFIBUS DP is used most often to operate sensors and actuators via a centralized controller in production technology. It also includes the connection of the networking of multiple controllers to one another.Data rates up to 12 Mbit/s on twisted pair cables and/or fiber optics can be achieved. PROFIBUS PA (Process Automation) is used to monitor measuring equipment via a process control system in process engineering and It is good for explosion-hazardous areas.. Here, a weak current flows through bus lines so that explosive sparks are not created, even for a malfunction. However data rate is only 31.25 kbit/s.

Profibus Architecture & Format

Profibus protocol uses the OSI layer model. It uses the application, data and physical layers. Various sevice levels were defined in application layer of the DP protocol ((**Ref.3).**

- DP-V0 In order to fullfil the cyclic exchange of data and diagnosis
- DP-V1 In order to fullfil the acyclic and cyclic data exchange and alarm handling
- DP-V2 In order to fullfil the isochronous mode and data exchange broadcast i.e slave-to-slave communication.

For the data link layer, it works in a hybrid access method that combines the token passing with a master-slave method. The controllers or process control systems are the masters and the sensors and actuators are the slaves in DP system. Various telegram types are used and are differentiated by their start delimiter (SD). **(Ref.3)**

SD2	LE	LEr	SD2	DA	SA	FC	PDU	FCS	ED

If there is no data then SD1 = 0x10, for Variable length data, SD2 = 0x68, for Fixed length data, SD3 = 0xA2, for Token, SD4 = 0xDC, for acknowledgement, SC = 0xE5
And SD=Start Delimiter, LE=Length of protocol data unit incl. DA,SA,FC,DSAP,SSAP), Ler=Repetition of protocol data unit, FC= Function Code, DA= Destination Address, SA= Source Address, PDU= Protocol Data Unit, FCS= Frame Checking Sequence, ED= End Delimiter.

For the physical layer, three different methods are specified. With EIA-485, twisted pair cables is used in a bus topology with data rates from 9600 bit/s to 12 Mbit/s. The cable length between two repeaters is limited to 100 to 1,200 m. This transmission method is used with PROFIBUS DP. With optical transmission via fiber optics, star-, bus- and ring-topologies are used. With MBP (Manchester Bus Powered) transmission technology, data and field bus power are fed through the same cable and are used in explosion-hazardous environments. The bus topology can be up to 1,900 m long with data rate 31.25 kbit/s. This technology was specially established for use in process automation for PROFIBUS PA.
In Profibus, manufacturer defines all device-specific parameters, functions and operating structures of his device with the help of a language similar to a programming language. Profibus is enhanced by Ethernet communication based standard PROFINET.

IEC 60870-5-101

The IEC Technical Committee 57 have developed a protocol standard for Telecontrol, Teleprotection, and associated telecommunications for electric power systems. The result of this work is IEC 60870-5. IEC 60870-5 provides a communication profile for sending basic messages between two systems. Five documents specify the base IEC 60870-5 (ref.4).

- IEC 60870-5-1 Tranmission Frame Formats
- IEC 60870-5-2 Data Link Transmission Services
- IEC 60870-5-3 General Structure of Application Data
- IEC 60870-5-4 Definition and coding of Information Elements
- IEC 60870-5-5 Basic Application Functions

IEC 60870-5-101 (IEC101) is an international standard which is prepared by IEC TC57 for power system monitoring, control & associated functions. It is completely based on the IEC60870-1 to IEC 60870-5-5 standards and uses standard asynchronous serial tele-control channel interface between both DTE & DCE. It is suitable for the multiple configurations like point-to-point, star etc.

IEC 60870-5 follows the EPA architecture.At the physical layer, IEC101 additionally allows the selection of ITU-T standards so the EIA standards RS-232 and RS-485 can be used and also support fiber optics interfaces. T101 specifies frame format FT 1.2, which is basically asynchronous and can be implemented using standard Universal Asynchronous Receiver/Transmitters (UARTs). Both fixed and variable block length formats are supported.

Data link layer defines whether an unbalanced or balanced transmission mode is used. For each link, address is also provided. IEC 870-5-2 specify that Send/No Reply, Send/Confirm, and

Request/Respond message transactions can be supported. It also defines the rules for devices that will operate in the unbalanced (multi drop) and balanced (point to- point) transmission modes. IEC 870-5-3 defines the Application Service Data Unit (ASDUs) for the application layer. Type information defines structure, type, and format for information object. Information elements in the T101 profile have been defined for protection equipment, voltage regulators, and for metered values. IEC 870-5-5 (1995) also defines initialization of station, Interrogation, Command transmission, Parameter loading, File transfer, Data acquisition using polling, Acquisition of events, synchronization of clock, Procedure of tests for T101 within the user layer.

IEC 101 also provides a checklist such as baud rate, link transmission procedure, basic application functions for the selection of the devices with respect to protocol perspective. IEC 60870-5 features supports unbalanced (master initiated message) & balanced (master/slave initiated message) modes of data transfer, Facility to classify the data into high priority (class-1) and low priority (class-2) and transfer the same using separate mechanisms.Cyclic & Spontaneous data updating schemes are also provided.

Comparison of DNP 3.0, IEC 870-5-101, Profibus and Modbus (ref.2)

Features	**DNP 3.0**	**Modbus**	**IEC 870-5-101**	**Profibus**
Standardization & Organization	Open industry standard (1993)-- DNP user's group	Medicon Inc.	IEC Standard (1995)- IEC TC 57 WG 03	BMBF(German department of education &research- 1989)
Layered Architecture	4-layer architecture -Also supports 7 layer TCP/IP or UDP/IP	Application layer messaging protocol	3-layer EPA architecture	3-layer architetcture
Physical layer	-**Balanced mode** -Supports multiple Masters & Slave, -peer-to-peer communication, -RS 232 or RS 485 Implementation, TCP/IP over Ethernet, 802.3	-**Balanced mode** of Transmission, -RS232serial interface Implementation, -Peer to peer Communication, -TCP/IP over Ethernet	-**Balanced Mode** -Point to Point & -Multipoint to point, -Implementation by X.24 & X.27 standard -**Unbalanced Mode** -Point to Point, -Point to Multipoint, -Implementation by V.24 & V.28 standard.	RS485/Fiber Optic -TCP/IP over Ethernet for PROFINET
Data link layer	Frame format FT3	Two message frame types: ASCII & RTU mode	Frame format FT 1.2	Various telegram type differentiated by SD
Device Addressing	Source and destination address (16 bits).	-Two characters (ASCII mode) or 8 bits (RTU mode) -Address range 1-247 -Broadcast address is 0	-Link address could be 0,1, 2 bytes -Unbalanced link contains slave address -Balanced link is point to	-0-127 addresses -127 for broadcasting -0-3 for master -126 for

			point .	general purpose
Configuration Parameters required	Baud rate Device addresses Fragment size	-Baud rate -Mode ASCII or RTU -Parity mode	-Baud rate -Device addresses -Balanced / unbalanced -Frame length -Size of link address -Size of ASDU address	-Baud rate -Device addresses -I/O channels -Diagnostic message
Dominant market	North America Australia and china)	application with low volume data in all over the world	Europe (South America, Australia and china)	With over 20 million devices all over the world
Open for other encoding solutions	Yes, like XML	Yes. source code like C, VC++ & JAVA etc.	Not Available	Not available

SPA-bus (ABB)

The SPA-bus was designed (proprietary protocol) as a fieldbus in a distributed protection by ABB to control and event reporting system. The system was developed to provide a communication between the slave units as protective relays and control units connected over the SPA-bus to a master unit. The physical implementation can be a fiber optic bus loop. Other media can also be used. SPA bus uses an asynchronous serial communications protocol with 1 start bit, 7 data bits with even parity, and 1 stop bit. The data rate with data transfer rate of 9600 b/s is possible. Data rate of 300/1200/2400 or 4800 bit/s can also be used. Messages on the bus consist of ASCII characters. The bus can support one master and several slaves.
The basic construction of the protocol assumes that the slaves do not request for the data themselves but the master is aware of the data which is present in the slaves and master can request that data. The master requests slave information using request messages and sends information to the slave in write messages. Master can also broadcast message to slaves.

Message formats
The messages contains ASCII characters (0AH, 0BH, 10H, 7EH). Those messages which are sent by the master are started with a begin bracket (">"), and the messages sent by slave start by characters("lf<"). The master messages end with character ("cr") (0DH) while the slave messages end with characters "crlf" (0DH and 0AH). In order to show the continuation of the message, the character "&" is used by the slaves. The maximum total length of the message is limited to 255 characters. There is no restrictions for the number of lines as far as the message maximum length is not exceeded. The master message has the following format:
(>nTe/eXm/m:xxxx/xxxx/xxx/xx:CCcr)
where > is the start character, nTe/eXm/m is header, xxxx/xxxx/xxx/xx is the data part, CC is the check sum, cr is stop character.
The slave message format is as follows:

lf<nT:xxxx/xxxx/xxx:CCcrlf
where if< is the start character, nTi s header, xxxx/xxxx/xxx is the data part, CC is the check sum, crlf is stop character. Where n = slave number, T = message type code, e = channel number, e/e = first/last channel, x = data category code (data type), m = data number, m/m = first/last data.
The delimiter character ":" separates the message header from the data part and the data
For example, Setting 1 of channel 1 of slave 2 is requested by the master using message:
>2R1S1:XXcr
The slave replies the request by message:
lf<2D:10.1:XXcrlf **(ref.6)**

Communications between master and slaves
The master initiates the communications by sending a message to the slave. The slave which recognizes the message for him responds by sending reply message. In case of error, slave either does not respond at all. So the master resends its message again. Those messages which are sent by the slave are not replied with an acknowledgment by the master. If the master detects an error in the message sent by the slave, the master resends its message. The communications can be divided into two categories as data reading from slave and data writing to slave.

UCA (Utility Communication Architecture)

In order to have a uniformity between all vendors devices, the first project was Utility Communications Architecture (UCA) project in the early 1990's, so that utility can use different vendor's equipments. Instead of creating a new protocol, UCA makes use of several existing international standards and technologies that the computer industry is using. UCA allows the open architecture & facilitates the use of distributed measurement, control, and communication (MCC) schemes. UCA 2.0 compliant does not mean interchangeable but however interoperable. So one cannot simply change the one vendor's device with another one until modifying the network.
Data transmission within UCA 2.0 is based on the Open Systems Interconnection (OSI) seven-layer model i.e using the existing protocol stack. Application-level data (MMS) from a device first enters at level seven, goes through all layers to layer one, travels across the medium (i.e., wire) to the receiving point, then makes it ways up all seven layers on that end. These layers can be grouped into three sets of profiles.
A-Profile includes the application, presentation, and session layers which provide the MMS functions. These layers decide how the information is to be packaged and in what form it will be sent. T-Profile specifies the transport layer and the network layer. The transport layer is responsible for ensuring reliable message transfer across the network. The network layer is responsible for providing addressing and routing functions. L-Profile is the data-link and physical layers of the stack. The data-link layer is responsible for controlling access to the network media. The physical layer specifies the actual physical media on which the data will be carried i.e copper, fiber etc.

STANDARD FOR ALL
IEC 61850
All of these existing protocols (open or proprietary) like PROFIBUS (GE), SPABUS(ABB), KBUS(ALSTOM) and the Common protocols are DNP3.0, MODBUS & IEC 60870-5 are not suitable to use with different manufacturer devices and a converter is required for them to communicate with each other increasing a huge engineering cost. So there was a need of the interoperability and long term stability between the communicating devices belonging to the different manufacturers.

The International Electro technical Commission (IEC) created a global and future proof standard IEC 61850 to meet all these requirements. IEC 61850 is basically a UCA 2.0 plus. The standard documentation contains 10 parts. It Standardizes data models for all application and uses the strengths of the OSI 7 layer communication model. For time critical data, the standard defines the data to be routed from application to data link layer directly so decreasing the transfer time i.e GOOSE messages. However sampled values are routed through whole of the OSI stack.

Substation Automation functions are separated from the communication stack by the ACSI (Abstract communication interface) In a real implementation this interface is mapped to an existing communication stack (OSI Layer), realized by the specific communication service mapping (SCSM). IEC 61850-9 specifies the mapping for the transmission of the sampled values and IEC61850-8 specifies the mapping for the all other communication services including the transmission of the station wide events. The data models are divided into logical groups (13 logical groups) called devices, nodes, classes and data. Each functional element is defined as a logical node (92 logical nodes). A physical device (IED) can house multiple logical nodes in it.
Each logical node is a collection of standard data classes (355 Data Classes). The possible values that can be assigned to the data classes are called as data. So in this way data are managed inside the IED's. These IED's are collecting data from the utility and communicating with the Master'' Station. The defined language is called Substation Configuration description Language (SCL). The IED and communication system model in SCL is according to IEC 61850-5 and IEC 61850-7-x. SCSM specific extensions rules .The configuration language is based on the Extensible Markup Language (XML) version 1.0. So the devices are able to transmit their configuration in XML upon request and visibility of the data provided by the vendor is available.
The new IEC61850 standard opens a vast opportunity to monitor and control substation automation functions and cost effective solutions are possible with the Existing IEDs. The Interoperability and Flexibility of integrating different manufacturer products will make this standard in practice very soon.

Conclusion

In order to choose which protocol should be used, it depends on the utility. IEC 870-5-101 and DNP 3.0 are mainly used in Utilities, Oil & Gas and power Industries. DNP 3.0 sends small number of large sized data while IEC 870-5-101 sends large number of small sized data. However Modbus is mainly used for Industrial applications where the amount of data transfer is not large. It is a very fast and safe protocol and lot of information is packed in one message. For communication within substation, the most common protocols (Open) is Modbus, , Profibus, UCA, IEC 61850 and some of the proprietary protocols are SPA (ABB), VDEW by Siemens, K-BUS by Alstom. For Communication outside the substations, protocols are IEC 870-5-101 &104, DNP 3.0, IEC 60670-6. For the Communication between applications the standard which can be used is IEC 61968.

References

[1] UCA 2.0: The GE Experience, Mark Adamiak Drew Baigent, GE Power Management King of Prussia, PA Markham, Ontario

[2] Comparison of protocols used in remote monitoring: DNP 3.0, IEC 870-5-101 & Modbus Jay Makhija (03307905), Supervisor: Prof. L.R.Subramanyan

[3]profibus-wikipedia

[4]IEC60870-5 -wikipedia

[5] Modbus and DNP3 Communication Protocols, Provided by: Triangle MicroWorks, Inc. Raleigh, North Carolina, Phone +1 919-870-5101 , www.TriangleMicroWorks.com

[6] Technical description SPA-bus, Communication, Protocol V2.5, Issued 96-01-23, Modified 2001.07.31, Version C, Checked MP, Approved TK

[7] Fundamentals of Utilities Communication Architecture, CAP T U T O R I A L.

[8] A basic explanation how modbus works, http://www.simplymodbus.ca/FAQ.htm