IEEE UCA™ AND IEC 61850 APPLIED IN DIGITAL SUBSTATIONS

Karlheinz Schwarz
Schwarz Consulting Company, SCC
schwarz@scc-online.de

1 Summary

World-wide, electric utility deregulation is expanding and creating demands to integrate, consolidate and disseminate **real-time information** quickly and accurately within and with substations. Utilities spend an ever-increasing amount for real-time information exchange; costs for **system integration** and **data maintenance** are exploding. **Development** and **sharing information** among all industry participants on critical resources (e.g., critical T&D equipment) is one of the most crucial issues for reliable power systems. In response to this need, IEEE has published a suite of international standards in the "Utility Communications Architecture (UCA™)" – IEEE Technical Report 1550 (1999).

UCA's objective is to dramatically improve **device data integration** into the information and automation technology, **reducing the costs for engineering**, **commissioning**, **operation**, **monitoring**, **diagnostics, asset management**, and **maintenance** and increasing the agility of the whole life cycle of a substation. UCA differs from most previous utility protocols in its use of **object models** of devices and device components. These models define common data formats, identifiers, and controls, e.g., for substation and feeder devices such as switches, voltage regulators, and relays. The models specify standardised behaviour (interoperability) for the most common device functions.

The standards applied in UCA (e.g., Ethernet, TCP/IP, and MMS) define and exchange **real-time data** and **metadata**. Some 3,000 standardised objects with their names and types are specified. The metadata can be used for **configuration** and **on-line verification** of the integration and configuration of databases throughout the utility. This **self-description** of the data significantly reduces the cost of data management, and reduces down time due to configuration errors.

The UCA models, services, and protocols for substation devices are currently being integrated into the drafts IEC 61850 (Communication networks and systems in substations).

This paper gives an overview on utility's crucial integration requirements, the IEEE UCA (IEC 61850) solution, and the global market acceptance of this new technology. Finally the feasibility of the coexistence of a UCA real-time communication server and Web, Telnet, and ftp servers running under an embedded LINUX are shown.

2 Benefits of the new approach

The Utility Communications Architecture (UCA) was developed under the sponsorship of the Electric Power Research Institute (EPRI, Palo Alto, USA). The objective of UCA is to provide seamless integration across the utility enterprise using off-the-shelf international standards to reduce costs. UCA Version 2.0 has been published as IEEE technical report TR1550 in November 1999.

UCA differs from most previous utility protocols in its use of **object models** of devices and device components. These models define common data formats, identifiers, and controls for substation and feeder devices such as measurement unit (see figure 1), switches, voltage regulators, and relays. The models specify standardised behaviour for the most common device functions, and allow for significant vendor specialisation for future innovation. The models have been developed through an open process including broad vendor and utility participation. These standardised models allow for multivendor interoperability and ease of integration. Modern protocols (such as those found in UCA) make use of the reduced bandwidth costs and increased processor capabilities in the end devices to carry metadata: standardised names and type information for the most common device information which can be used by applications for on-line verification of the integration and configuration of databases throughout the utility. Examples for measurement metadata are "unit", "offset", "scale", "dead band for reporting", and "description". This feature significantly reduces the cost of data integration, data management, and reduces down time due to configuration errors.

The UCA object models are defined in terms of standardised types and services. These services (such as reporting by exception and select before operate controls) are defined in abstract terms, then mapped to messages in the underlying application layer protocol. UCA Version 2.0 application layer services for data acquisition and control functions in all profiles are provided by ISO/IEC 9506: Manufacturing Message Specification (MMS). The use of the standardised service definitions above MMS allow for 'future-proofing', in that new innovations in application layer protocols can be incorporated into future versions of UCA without disturbing the object model definitions.

The MMS protocol, developed by the manufacturing community, supports real-time control and data acquisition. MMS defines a message structure supporting access to data, programs, journals, events, and other constructs common to real-time devices. These messages may be transported using many different underlying protocol stacks.

3 Utility Communications Architecture (UCA)

The UCA documents specify a set of existing international standards which can be applied to specific communications architectural requirements in the utility industry. UCA can be used to define and implement a wide variety of standards-compliant communications systems such as those required to support Distribution Automation, Demand Side Management, Substations and Control Systems, Power Plant Automation, and Customer Interfaces.

UCA comprises the following documents:

Common parts

- Introduction to UCA
- UCA Communication Profile Specification

Modeling and communication for intelligent devices

- Common Application Service Models (CASM),
- Generic Object Models for Substation and Feeder Equipment (GOMSFE),
- Customer Interface Device Models (under preparation)
- Power Plant Device Models (under preparation)

Real-time data exchange between control centers

- IEC 60870-6-503: TASE.2 Services and Protocol
- IEC 60870-6-802: TASE.2 Object Models
- IEC 60870-6-702: TASE.2 Application Profile

Many parts of the UCA have been also progressed through the standards process as IEC international standards. The UCA approach to communication between control centres, power plants, and SCADA masters was developed as the Inter-Control Centre Communications Protocol (ICCP). ICCP was later taken up by IEC TC57 working group 7 and standardised as IEC standards 60870-6-503 and 60870-6-802 (TASE.2). These standards define methods for using MMS to synchronise databases, as well as to perform scheduling, accounting, and other messaging.

The UCA Version 2.0 models, services, and protocols for substation devices are currently being used as the basis for IEC 61850 (Communication networks and systems in substations). The committee drafts of IEC 61850 will be distributed to IEC member countries for balloting by end of 2000.

The UCA comprises the data object models (forming the highest level), the service interfaces to these models (defining, retrieving, reporting, and logging of process data, controlling devices, file transfer etc.), and the communication profiles (see figure 2).

UCA Communication profiles

Similar to current Internet solutions, UCA provides a network solution to interconnect data sources within and between utilities.

Ethernet was chosen as the main solution because of its:

- Market dominance;
- Plentiful, low-cost hardware, such as bridges and routers; and a
- Scalability from 10, 100, and 1000 Mbit/s, with 10 Gbit/s becoming available soon.

The UCA makes use of a family of international protocols, organised according to the Open Systems Integration (OSI) reference model. The reference model allocates the communications functions to defined layers, then supports a variety of standards at each layer to allow for various price and performance options. Each industry sector then chooses from the options at each layer to define one or more profiles. The UCA includes two primary 7 layer profiles, one using OSI standards and the other TCP/IP. The UCA also includes a 3 layer profile for use over serial links in low-cost devices (see figure 3).

Common Application Service Model (CASM)

The UCA Common Application Service Model (CASM) provides a common set of communication functions for data access, reporting, logging, control applications and related support. The use of a common set of services allows for 1) isolation of the models from service and communication details, 2) a high level of application interoperability, and 3) reduced integration and development costs through the use of common mechanisms for data access and communication establishment. The CASM services are abstract and may be mapped to existing communication application standards. MMS (ISO 9506) is the service specification of choice. Mapping of CASM to MMS is included in the UCA document

Generic Object Models for Substation and Feeder Equipment (GOMSFE)

One of the primary tasks has been the development of models for protective relay functionality along with all other anticipated IEDs in the substation. The development of these IED models is known as the Generic Object Models for Substation and Feeder Equipment (GOMSFE). Starting with a base set of models, each of the relay vendors has added draft models for an additional one or two functions, which brought the total to 13 models. These 13 protective relay function models have been reviewed in depth, and two basic building block models were developed (Basic Relay Object and Basic Time Curve Object). The existing models have been reworked to use the basic building block objects, and add extensions as necessary. It was concluded that an additional 20+ relay types could be modelled using the basic building blocks.

An excerpt of GOMSFE device models are listed below. These models define some 3000 tagged information like vendor name, software revision, switch position status, current phase A measurement, or control a switch.

Excerpt of the UCA functional models:

- Generic Input/Output
- Measurement Functions
- Transformer Functions
- Switch Functions
- Reactive Functions
- Protection Functions
- Distance (DIST)
- Synchronising or Synchronism-Check (SYNC)
- High Impedance Ground Detector (HIZR)
- Directional Overcurrent (DOCR)
- Reclosing Relay (RECR)
- Differential Relay (DIFF)
- Measurement Unit
- Basic RTU Object Models
- Transformer Object Models
- Switch Object Model
- ...

These object models provide the interoperability of the various devices and systems connected in substations. They define the semantic of operations.

Figure 4 shows the relation of UCA and IEC 61850 documents.

4 UCA substation demonstration initiative

EPRI's UCA substation demonstration initiative project led by AEP (American Electric Power) has as its goal to produce industry consensus regarding Substation Integrated Control, Protection and Data Acquisition, and to allow interoperability of substation devices from different manufacturers. To this end, an open process has been followed on this project, to review each major project document and milestone in the open forum of standards-related organisations. The initiative is an excellent opportunity to present the benefits of the (redundant) Fast Ethernet and the device modelling technology.

The UCA 2.0 profiles for field equipment communications are separated into Application Profiles, Transport Profiles, and Data Link Profiles. These profiles are combined to form complete Profiles that can meet different requirements.

By adopting existing standards, the utility can take advantage of the economies of scale of the electric utility and industrial control industry that has made extensive use of these protocols. The substation initiative is now supported by some 30 utilities and 25 Substation device and systems vendors.

A list of vendors can be found at:
www.nettedautomation.com/solutions/uca/products/vlist/index.html

5 Application in the gas industry

UCA was adapted by GRI (Gas research institute, USA) for use by gas utilities. This effort culminated in an evaluation of UCA in a gas utility environment at Pacific Gas and Electric Company, San Francisco. With gas industry operations becoming more complex, as the study shows, the benefits of UCA are significant. With UCA in place, system operators can more easily automate systems, gather operating data, exchange information, and analyse historical statistics.

The benefits of UCA include:

- The enhanced ability to develop integrated business applications across functional areas.
- Simplified implementation of fully integrated communications networks.
- Purchasing alternatives from multiple vendors for compatible hardware and software.
- Reduced operating costs through reductions in installation, maintenance, operation, and training.
- An enhanced ability to respond quickly to the continuing changes of a less regulated, more competitive business environment while still offering value-added customer services.

At Pacific Gas and Electric Company, UCA-compliant equipment was used to collect distribution system data (e.g., pipeline pressures, flow rates, and gas quality) at regulator stations and throughout a distribution piping system, along with information on customer load, weather, cathodic protection, and other conditions. The estimated cost savings demonstrated in the field experiment, extrapolated to the gas industry as a whole, is $133 million, with the potential for an additional $47 million savings ($180 million total) by further integrating and consolidating data collection and monitoring functions into a single "intelligent electronic device" at field sites.

6 TASE.2 (ICCP) globally adopted

An early TASE.2 adopter in the United States, the New York Power Pool (NYPP), completed implementation of a TASE.2-compliant communications system. A consortium of the seven investor-owned utilities of New York state and the New York Power Authority, the NYPP was operating a proprietary communications protocol that had limited capabilities. NYPP recognised that a standardised communications protocol that expanded the pool’s capabilities and enabled real-time exchange of data would best serve its members in the changing business environment.

Because of the TASE.2 protocol’s standardised nature, the NYPP can now utilise the most advanced telecommunications technologies, such as frame relays and ISDN lines, to expedite data transmission. The lower initial cost of the TASE.2-compliant system, compared to a proprietary system, provided immediate saving - estimated at $300,000. In addition, the pool’s recurring communication costs, such as telephone charges, will be cut in half, saving NYPP an estimated additional $780,000 over five years. Moreover, the system will also provide a communications gateway into the United States for Hydro Quebec, one of the Northeast United States’ major power providers.

These savings are typical of the early adopters of TASE.2 in the United States during 1995 and 1996. In 1997, competition and standardisation reduced TASE.2 system costs even more – by as much as a factor of four! This price reduction occurred as vendors sold TASE.2-compliant systems as a fully developed standardised product. “Further reductions of more than an additional 40% are feasible in 1998 and beyond,” says EPRI’s David Becker, “as computer hardware costs decrease and communications system software is increasingly run on relatively low-cost operating systems such as Windows NT.”

The collaborative efforts that have produced the TASE.2 standard are reaping rich rewards for energy companies world-wide. The protocol has gained widespread acceptance over the past year, with numerous vendors offering TASE.2 products. There are an estimated 250-300 completed or current implementations of TASE.2-compliant systems world-wide (1999).

Without TASE.2 utilities would need to establish a variety of independent grow-as-you-go, point-to-point links. Since all major EMS vendors have adopted TASE.2, utilities can use the same protocol to communicate between all of their control areas and members, regardless of the EMS equipment they use.

7 Re-usability and device modelling

Describing device functionality by specifying the data (syntax and semantic) and the dynamic behaviour (state machines) of devices (as seem from remote) is one of the crucial challenges in the standardisation. Many standardisation groups have started defining different views of domain-specific device types. The views are e.g.:

- Engineering (in the context of a plant),
- Commissioning,
- Configuration,
- Operation,
- Maintenance,
- De-commissioning

Hardware and software, as well as communication networks are subject to frequent innovation. Therefore, it is worth-while to standardise independent (abstract) interfaces for communication networks and the access to the application objects.

The abstract application objects (objects define the semantic of the device functions) will continuously be used (with minor changes only). The object definitions will be enhanced in the future to meet additional requirements, i.e. re-using the definitions specified in the past (see figure 5).

The most important objective of the device description is to define re-usable parts to be used for specifying the data models and behaviour of various types of industrial devices. Re-usability has two aspects. First, re-use of a given functionality in many devices throughout an application domain (we may call this: horizontal re-use). Second, re-use of a given function in the definition of an enhanced or specialised function (we may call this: vertical re-use). The re-usability is a crucial factor in reducing the costs of the overall system design, engineering, operation, and maintenance. Support of re-usability is the key issue in the standardisation!

8 Benefit of device modelling

The real benefit of device modelling is the re-use of (common) definitions made in the past. This is our daily practice! We are using common terms at work (key board, laser printer, office, ..) or at home (kitchen, chair, wheel char, bath room, ...). Just misunderstandings are the result if terms are not understood uniquely on both sides (sender and receiver). It is not only a matter to define something completely – more important is, to understand it uniquely. All technical specifications in the area of distributed systems have to follow distinct rules for defining and exchanging information, and unique interpretation of this exchanged information.

Interpretation is quite easy if we can re-use common terms learned in the past. In our daily life we re-use (instantiate) the term “laser printer” (more precise we re-use the class definition that is associated with term “laser printer”) for a laser printer next to you “laser printer in room 23” or we may re-use the term for a special type of a laser printer: A4 laser printer (“A4 laser printer in room 23”).

Distributed systems should operate in the way they have been told to do. If they do not? This may have many reasons. A major issue is, that independently developed devices may follow the specification of their implementers but the implementers may have different interpretations of the specification that describes the co-operation of the devices!

Devices will not operate in the way they should do, if the human beings (the implementers) do not understand each other!

Device models are collections of terms with associated semantics and a description of the dynamical behaviour. As an example of modelling the switch controller of IEC drafts 61850-7-x (Communication networks and systems in substations) or IEEE TR 1550 (Utility Communications Architecture, UCA) is shown and discussed next.

The model definitions shown in this article are incomplete; the objective is to discuss the principles only.

The switch controller model is defined as a set of attributes which are inherited from the switch class or defined in the switch control class (see figure 6).

The class “Sw” (switch) defines a simple base class with less than ten attributes. The class “SwC” (switch controller) uses the attributes of the switch and adds some other attributes specific for the controller of the switch.

The box in which the switch controller is located is called a “virtual device”. Usually models are abstract in the sense that they do describe only those aspects that are visible to the remote user of a device. It is sufficient to know the external visible data and behaviour of the device (the WHAT). The concrete realisation of the device, its internal interfaces and programming language or operating system (the HOW) are not of interest for the view from outside. To understand the concept of a virtual system, the following saying may help.

If it's there and you can see it	It's REAL
If it's there and you can't see it	It's TRANSPARENT
If it's not there and you can see it	It's VIRTUAL
If it's not there and you can't see it	It's GONE

Roy Wills

The list of the (virtual) attributes of the two classes are depicted in figure 7.

The switch controller class “SwC” (this abbreviation "SwC" is defined in the standard) has many attributes. Three of them are inherited from the primitive switch class “Sw” on the right hand side. An instance of the “SwC” may be referenced as “SwC5”. All attributes of the class SwC fall into specific categories (Functional components, FC) like: “MX”, “ST”, “CO”, “CF”, “DC”. These terms (abbreviations) indicate a specific semantic of an attribute. “MX” stands for Measurements, “ST” for status, “CO” for control, “CF” for configuration, “DC” for description.

All attributes of the class are named (object name). Each object name carries a semantic, too: “OperCnt” has the semantic “Number of switch operations”, “SwDS” represents “Indication of the position of the switch of type device status (DevSt)”.

NOTE – The Data types and the value range of the attributes are not shown here, they are defined in the class definitions as well.

These names are used to define, exchange, archive, or access the data dictionary of a real switch controller of class “SwC”. The switch position of switch #5 is referenced by the following concatenation: “SwC5.ST.SwDS”.

The concrete switch controller #5 is defined as a (hierarchical) list of attributes that make up the data dictionary of that specific switch controller #5:

SwC5.MX.OperCnt

SwC5.MX.SwOperTim

SwC5.MX.CtlCmdCnt
SwC5.ST.SwDS
SwC5.ST.LocRemDS
SwC5.ST.CtlFailInd
SwC5.ST.CtlTagBlk
SwC5.ST.CtlIntlkBlk
SwC5.CO.ODSw
SwC5.CF.ClockTOD
SwC5.CF.OperCnt
...

This complete list of attributes (with all the names) is defined in the standard – with the exception of the number "5" that indicates the switch controller number "5".

The re-use of the switch controller class "SwC" is as easy as copying some lines of text and add the instance specific information, e.g. instance # "5".

The class "SwC" is part of a repository for substation device models. The repository holds a list of various (standardised) classes that can be used as templates. Real devices can now be build compliant with these classes.

The total number of attributes of all UCA objects is some 3000 (flat) data points.

Attributes are often derived from other classes. Many attributes are inherited from common classes. Common classes are composed out of some 150 common components, e.g., q (=Quality) or s (=Scale).

These names, their semantic and their types are used to build the device classes of IEEE TR 1550 Volume 2 – Part 4 (GOMSFE – Generic Object Models for Substation and Feeder Equipment). This document could be understood as a class repository for substation and feeder equipment. The repository is a source of classes to be used to construct simple and complex devices.

About all attributes and classes of TR 1550 will be specified and published in IEC 61850-7-3 and 61850-7-4. IEC TC 57 WG 10-12 members will define additional models and attributes.

The example device modelling as shown above provides re-use of structured information (semantic) for:

- Definition of device classes based on other classes (new class inherits attributes of base classes; re-use of base classes); this allows to define vendor- or user-specific classes that are specialisations of available classes,
- Instantiation of classes (instances inherit the class attributes; re-use of classes),
- Messages (instances) inherit the name structure from message classes (e.g. Control, Report) and from function classes (e.g. "SwC"); re-use of message classes and name structures.

This comprehensive model allows for seamless engineering and remote access. The remote access can be applied for operation, configuration, maintenance, and other functions.

Many (hierarchical) application names simply pop up when class models defined in IEEE TR 1550 (and IEC 61850) are instantiated during system configuration.

The system engineer does not need to care about the structure and naming convention and hierarchical names – they are all pre-defined in the standard and can just be re-used. He can learn the structured names once, and apply them many times - independent of the vendor!

The re-usability is the biggest keyword in the standardisation of models for industrial automation systems. The approach of TR 1550 and IEC 61850 reduces the efforts of engineering and operation dramatically – thus saving a lot of resources.

In the past, configuration of several systems from different vendors or from different system families led to the situation that the engineers had to learn as many system structures (terms, semantics, and syntaxes) as they use different systems.

Just a fraction of this total cost has been spend in the process of defining, commenting and refining the definitions in the standardisation so far. This process has taken several years and has cost man-years of efforts of many domain experts (under the umbrella of IEEE, IEC and EPRI) to reach world-wide consensus.

Without the bunch of classes the cost of defining a well structured system (that is accepted by a vendor and several users) would be much higher than using a well structured "template" and use it again and again.

9 UCA™ in-a-LINUX-box successfully demonstrated

A feasibility study showed that UCA can be easily integrated into an embedded system. The results are very positive with regard to memory needed and functionality provided. The following has been demonstrated in January 2000:

- Sub-Credit card size PC
- JUMPtec DIMM-Concept
- Embedded LINUX (EMJ-Linux derived from Slackware-Linux)
- Embedded UCA (IEEE TR1550)
- Embedded Web, ftp, and telnet server
- 10 Mbps Ethernet (2nd DIMM board today)
- TCP/IP
- Example Web site

Two GOMSFE models have been implemented:

- MMXU (Measurement Unit) some 220 Data Objects
- PBRO (Basic Relay Object) some 125 Data Objects
- Total number of MMS Variables some 350

The most important MMS/CASM Services implemented are:

- Initiate
- Indentify
- GetNameList
- Status
- Get Capabilities
- Read
- Write
- GetVarAccessAttributes
- GetNamedVariableListAttributes
- GetDomainAttributes
- FileOpen
- FileRead
- FileClose
- FileDelete

- FileDirectory
- UnsolicitedStatus
- InformationReport/Reporting
- Conclude
- Cancel
- Abort

The UCA Server comprises the following components running under LINUX:

- RFC 1006 (Mapping to TCP/IP)
- Session
- Presentation
- MMS encoding/decoding
- MMS services
- UCA CASM services
- UCA GOMSFE Models (some 350 domain specific variables)
- Runtime scheduler (Test Application)
- Data value simulator (e.g. controls changes of data)

All components are compiled and linked under LINUX GNU compiler:

The software size of the EXE file is **just 205 Kbyte**. Compared to the software size of the EXE file under Windows (compiler specific) which is 350 - 450 Kbyte.

<u>Possible Applications of UCA in-a-LINUX-box are:</u>

- Gateways
- Retrofit Kit (saving costs for the installed equipment)
- Embedded into IEDs (Intelligent Electronic Devices)
- RTUs (Remote Terminal Units)
- Tele-control
- PLCs (Programmable Logic Controllers)
- PCs (Personal Computers)

Additionally the UCA in-a-LINUX-box hosts a web server (Apache), a ftp server, and a telnet server. A mapping of a relay model to a XML file on the server is shown on figure 8. The XML file represents the relay model for easy non-real-time access by a web browser. Intelligent devices will have these two stacks: UCA for real-time and web technologies for non-critical communications. The device and data models are the key issue in UCA and IEC 61850. The mapping to appropriate solutions depend on the requirements and on the technologies available.

More details see:
www.nettedautomation.com/solutions/uca/linux/uca_linuxbox/inaLINUXbox_01.html

10 Retrofitting installed equipment with seamless communications

The maintenance, integration, and operation cost for installed systems are high at the beginning of their life cycle. Some time down the road they reach a minimum and increase dramatically after the minimum has passed. The reasons for the increase are manifold and well known. For example, the software used will not be supported and updated any more after let's say ten years, additional information (for maintenance, diagnostic, asset management, ...) may not be accessible even if it is available locally in the equipment's computer system. These systems may not be replaced within the next decade or two!

How can this source of useful information be made available (opened) to any system in the enterprise that needs this information? Today, we see coming up hundreds of solutions to make this information available. Almost all of them are proprietary and do not support a seamless integration.

Until today, the priority was on a high-speed network "copying" cyclically a small fraction of the available information on the network for real-time control. Now, enterprise applications demand more information accessible at the system, and: reduced integration cost, more advanced technology, more productivity. To meet these objectives, you need be able to keep track, see, understand, analyse, and adjust to what happens on the plant floor. The enterprise needs to access real-time information seamlessly between the process level and higher-level systems.

System specific information of proprietary systems can be modelled the very same way as standardised models are defined. The CASM services and the common definitions can be re-used, too.

11 Summary

Deregulation will place greater demands for information on utilities than they have experienced before. Development and sharing information among all industry participants on critical resources (e.g., critical T&D equipment) is one of the most crucial issues for reliable power systems. IEEE's UCA TR 1550 (IEC 61850) provides a timely, cost-effective, and standardised solution to allow advanced IED functions and distributed systems to form the foundation for 'next Generation' electric utility protection, control, and monitoring systems.

The benefactors of the results of open device data integration span the entire industry and include all of the stake-holders in this industry. The customers are in a position to save large sums of money and time. The vendors who provide solutions that meet or exceed expectations will become very successful. This is an exciting time in the industry with an inexorable move toward practical software components.

By providing a common communications protocol stack, UCA IEC 61850 allow an utility and other industries to “plug and play” equipment from different vendors. The specification of the uniquely tagged semantic of the most important device model data leads to a tremendous cost reduction during engineering, commissioning, operation, asset management, and maintenance.

An UCA/IEC61850/MMS evaluation CD ROM brings the world of the TCP/IP and OSI based open real-time systems according to UCA/IEC61850/MMS to your desktop! This is the world's first and only package that comes with everything included for the fastest possible start to learn about UCA/-IEC61850/MMS technology.

For details see: www.Nettedautomation.com/solutions/uca/evalkit/index.html

12 References

- IEEE Technical Report 1550 (1999): Utility Communications Architecture, UCA
- IEC 60870-6-TASE.2: Telecontrol application service element 2
- Committee drafts IEC 61850-7-y: Communication networks and systems in substations – Basic communication systems
- Becker, David L.; Becker, Gerhard; Gärtner, Kaib, Wolfgang; W.; Kimpel, V.; März, W.; Schmitz, W.; Schroeder, Robert; Schwarz, K.: Open Communication Plattforms for Telecontrol Applications – Benefits from the New Standard IEC 60870-6 TASE.2 (ICCP), etz-Report 32, VDE-Verlag Berlin, 1999.

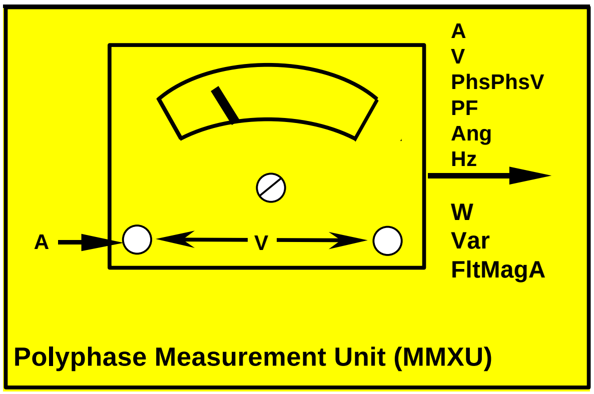
This article has been published by PennWell in the procedings of the the Distributech Europe 2000 conference, Vienna, 10-12 October 2000.

Figures:

GOMSFE

Chapter 7.4.1 Polyphase Measurement Unit (MMXU)

Polyphase Measurement Unit provides for measurement of single phase or polyphase analog values (including neutral), pertaining to a wye or delta connected field device or circuit.



FC	Object Name	Class	rwec	m/o	Description
MX	V	WYE		o	Voltage on phase A, B, C to G
	PhsPhsV	DELTA		o	Voltage AB, BC, CA
	A	WYE		o	Current in phase A, B, C, and N
	W	WYE		o	Watts in phase A, B, C
	TotW	AI		o	Total Watts in all 3 phases.
	VAr	WYE		o	VArs in phase A, B, C
	TotVAr	AI		o	Total VArs in all 3 phases.
	VA	WYE		o	VA in phase A, B, C
	TotVA	AI		o	Total VA in all 3 phases.
	PF	WYE		o	Power Factor for phase A, B, C
	AvgPF	AI		o	Average Power Factor of all 3 phases.
	Ang	WYE		o	Angle between phase voltage and current
	Hz	AI		o	Power system frequency
	FltMagA	WYE		o	Fault Magnitude in phase A, B, C, N
CF	All PMXU.MX	ACF		m	Configuration of ALL included PMXU.MX
DC	All PMXU.MX	d		m	Description of ALL included PMXU.MX
RP	brcbMX	BasRCB	rw	m	Controls reporting of Measurements
AS	LogDev<n>	TBD		o	Defines path for Peer to Peer Communication

MX = Measurments, CF = Configuration, DC = Description, RP = Reports

Figure 1: Modelling Example

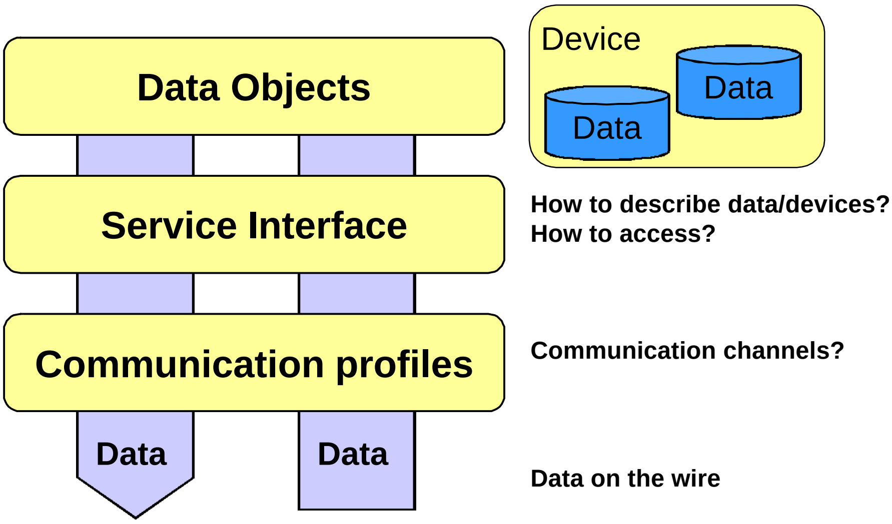


Figure 2: The three levels of UCA

	Full 7 CO	WAN 7 CL	Modified 7 CO	Reduced Stack CO	Reduced Stack CL	LAN-Based FAIS	LAN-Based ** Ethernet	TCP/IP RFC 1006	TCP/IP RFC 1070	TCP/IP RFC 1240
Application	MMS ACSE	MMS CL-ACSE	MMS ACSE	MMS ACSE	MMS CL-ACSE	MMS	MMS ACSE	MMS ACSE	MMS ACSE	MMS CL-ACSE
Presentation	Presenta-tion	CL Pres.	FastByte Pres.					Presenta-tion	Presenta-tion	CL Pres.
Session	Session	CL-Session	FastByte Session					Session	Session	CL-Session
Transport	TP4	CLTP	TP4					TP0 TCP	TP4 CLNP UDP	UDP
Network	CLNP	CLNP	CLNP			Auxiliary		IP	IP	IP
MAC Data Link	LLC1 ADLC FT3 or UCA 1	LLC1 ADLC FT3 or UCA 1	LLC1 ADLC FT3 or UCA 1	LLC1 ADLC FT3	LLC1 ADLC FT3 or Ethernet	LLC3 802.4 Token Ring	LLC3 ADLC FT3 ***over*** Ethernet	Ethernet SLIP, PPP (typical)	Ethernet SLIP, PPP (typical)	Ethernet SLIP, PPP (typical)

7 Layer (Full 7 CO – Modified 7 CO) | 3 Layer (Reduced Stack CO – LAN-Based ** Ethernet) | TCP/IP (TCP/IP RFC 1006 – RFC 1240)

Figure 3: UCA communications architecture

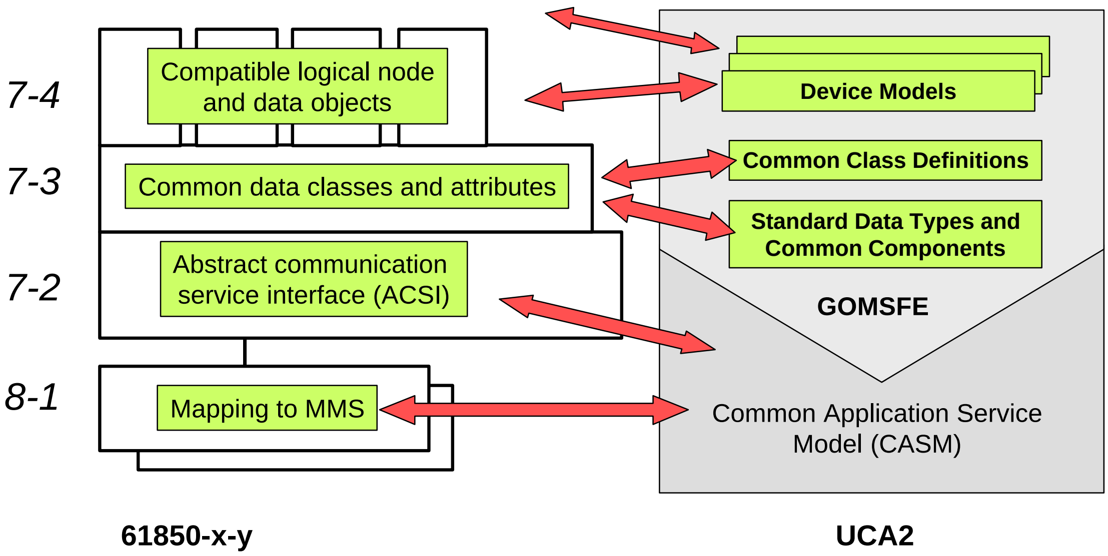


Figure 4: IEC 61850 / UCA documents compared

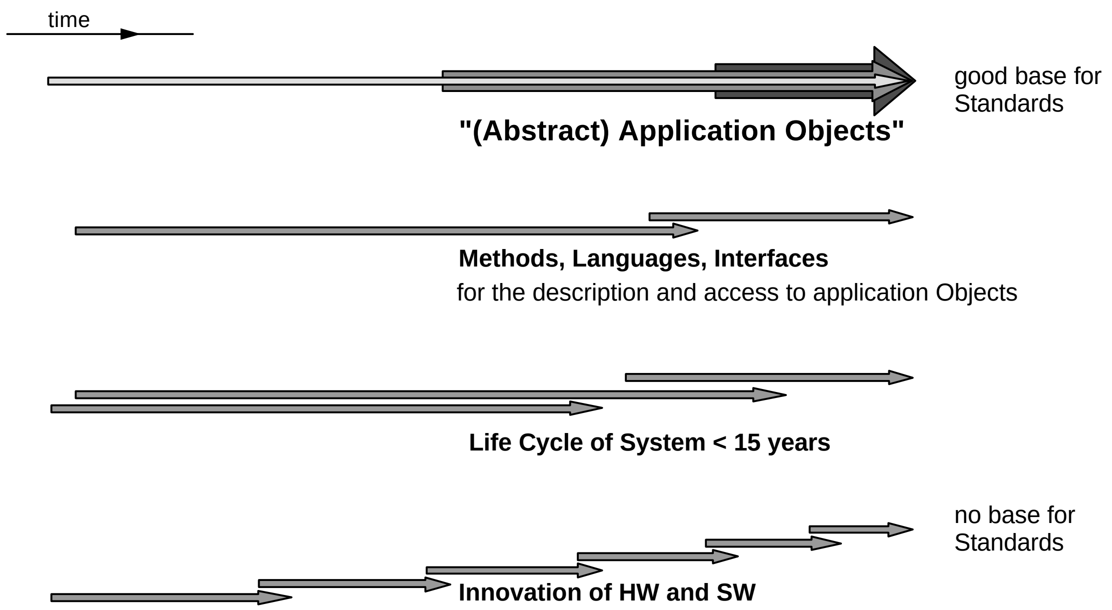


Figure 5: What is important to be standardised?

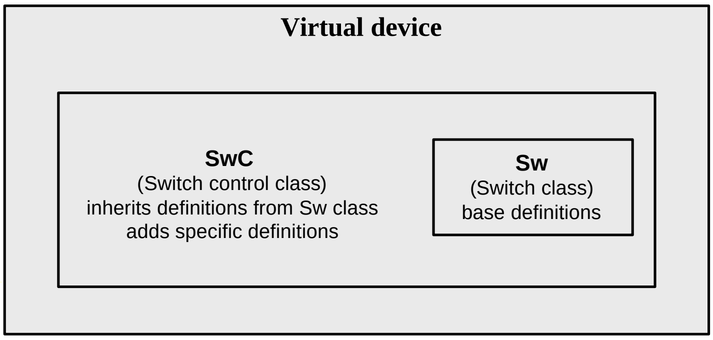


Figure 6: Switch controller and switch class

SwC (Switch control class)

FC	Object Name	Description
MX	OperCnt	Number of Sw operations
	SwOperTim	The Tim in msec for the switch to complete Oper from when the *ODSw* Cmd was issued
	CtlCmdCnt	The number of times that the controller has Cmd the *Switch* to Oper
ST	SwDS	*Inherited from Sw*
	LocRemDS	The mode of control, local or remote (DevSt)
	CtlFailInd	*CtlFailTime* has expired w/o the control action occurring.
	CtlTagBlk	*Switch* operation blocked due to a tag
	CtlIntlkBlk	*Switch* operation blocked by Intlk logic.
CO	ODSw	The command to open/close the switch.
CF	ClockTOD	Date and time for data logging.
	Same As SwC.MX	Configuration for SwC.MX
	Same As SwC.CO	Configuration for SwC.CO
	Img	Proprietary Information
DC	All Sw.DC	*Inherited from Sw*
	Same As SwC.MX	Description of all included SwC MX
	Same As SwC.ST	Description of all included SwC ST
	Same As SwC.CO	Description of all included SwC CO
AX	All Sw.Ax	*Inherited from Sw*
RP	brcbMX	Controls reporting of Measurements
	brcbST	Controls reporting of Status Points

Sw (Switch class)

FC	Object Name	Description
ST	SwDS	Indicates the position of the switch (DevSt)
DC	SwRtg	Description of the Equipment
	ConCkt	Description of the Circuit
	SwDS	Description of SwDS
AX	TagTyp	The allowable tags on the *Switch*
	Tags	The active tags on the *Switch*
	OpenIntlk	Opening of Sw prevented by interlock logic
	CloseIntlk	Closing of Sw prevented by interlock logic
RP	brcbST	Controls reporting of Status Points

Figure 7: UCA switch control class

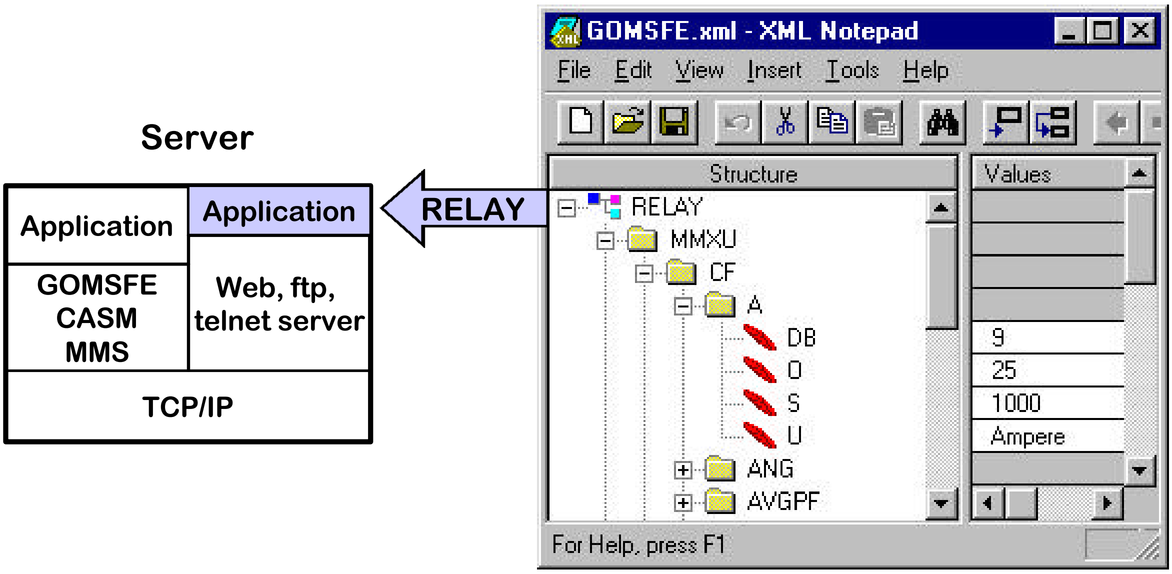


Figure 8: UCA models mapped to XML