IEEE UCA™ and IEC 61850 – SEAMLESS COMMUNICATION FROM POWERPLANTS TO CUSTOMER INTERFACES

Karlheinz Schwarz, SCC

1 Summary

Globally, utility deregulation is expanding and requiring demands to integrate, consolidate and disseminate **real-time information** quickly and accurately within all kinds of utility automation systems – from power plants to customer interfaces. Utilities spend an ever-increasing amount for real-time information exchange; costs for **data integration** and **maintenance** are exploding. 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 provides standards to dramatically improve **device data integration** into the information and automation technology, **reducing engineering**, **commissioning**, **operation**, **monitoring**, **diagnostics**, and **maintenance** costs and increasing the agility of the whole life cycle of utility automation systems. UCA differs from most previous utility protocols in its use of **object models** that model – most common – real devices and device components. These models define common data formats, identifiers, behaviour, and controls, e.g., for substation and feeder devices such as switches, voltage regulators, and relays.

The standards selected in UCA (e.g., Ethernet, TCP/IP, and MMS) make use of advanced IT solutions, the reduced bandwidth costs and increased processor capabilities in the end devices to define and carry **metadata:** more than 3,000 standardised names and type information which can be (re-)used by applications for **on-line verification** of the integration and configuration of databases throughout the utility. This **self-description** significantly reduces the cost of data management, and reduces system down times due to configuration errors.

This paper gives an overview on utility's crucial integration requirements, the IEEE UCA solution (including an overview on IEC 60870-6 TASE.2 – Inter-control center communication protocol, ICCP), and the global market acceptance of this new technology.

2 The challenge

Imagine if you didn't have common electric outlets and plugs in your house, and every time you bought a new appliance, you had to wire up the appliance to the wires in your wall. And everybody's wires in everybody's walls were different. And everybody's appliance wiring was different. That's really the way it works today with trying to integrate device data into applications and these devices into automation systems. Examples for device data are status, diagnostic information, measurements, configuration, description, and control information. This situation forces developers of application software and devices to write new drivers daily – implementing just new gateways!

Many utilities are faced the problem of islands of information based on proprietary technologies, each of which literally speaks its own language. The challenge is to **integrate all those**

islands of various applications into a functioning utility automation system (Figure 1). Control center for example need to know the overall operating conditions (gross load, plant activity, etc.) but the corporate culture is often resistant to telephone and fax communication, thus, information flow between facilities is limited.

Many utilities are using the UCA standard as a bridge between power plants, substations, and the control center, and to communicate within substations. They now have a broader perspective with more information on overall operating conditions such as change of loads, power production schedules, and other plant information.

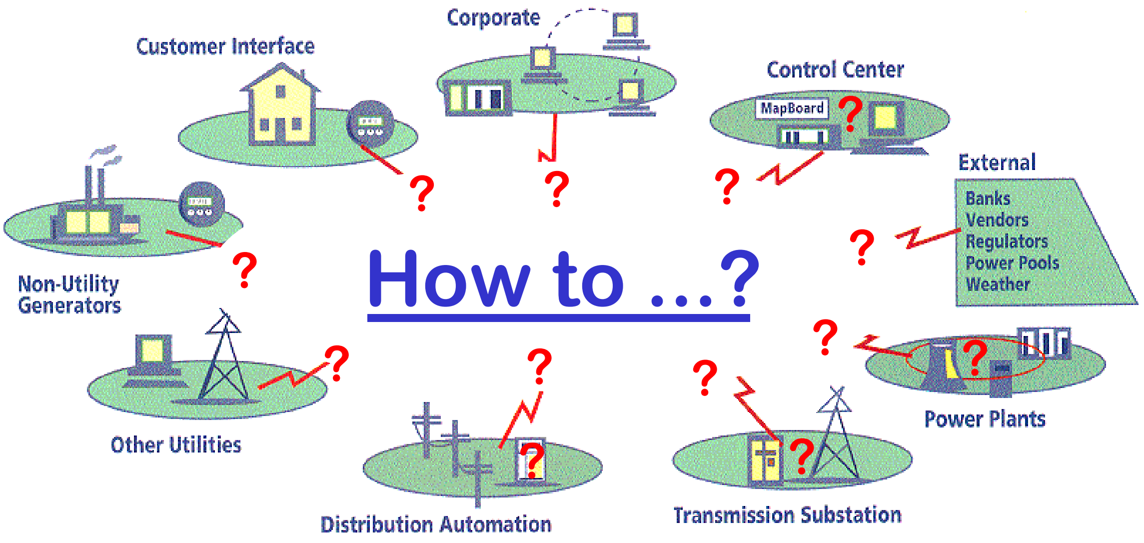


Figure 1 – Islands of information

An answer has been found in standards-based communication systems as embodied in **UCA**. To show just how important standards-based communication systems such as UCA are, consider the example of one large electric utility. At present, this utility has more than **200 different protocols running** on intelligent devices within its distribution network! A vendor is proud of supporting more than 100 different RTU protocols!

Industry experts say: **US $82 billion was spent on application integration in 1998** (= 40% of corporate IT budgets; still growing) – Forrester, 1999. Any 1 per cent of the Dollars spent for the integration of devices into applications costs some US $800.000.000 per year. How many per cent this integration costs is not known. The costs of integration are not well documented. Many companies do not keep specific records of the cost of integration. The total ongoing costs for system integration and maintenance for European utilities (some 220 Million US$) will be 50% higher than projected for 1999 (Source: Datamonitor).

The cost is not limited to installing new applications, customers explain. The larger expense in time and money comes from the overwhelming task of maintaining the API's to existing in-house applications. Many customers tell they believe that savings in maintenance alone is the biggest opportunity for saving time and money for an enterprise.

To reduce the risk of getting even worse in the future, the integration of – more and more intelligent – devices into the enterprise applications (SCADA, real-time asset, machine diagnostics, ...) is a real challenge for programmers and engineers.

The driving force behind the standardisation is to effectively and efficiently perform **seamless device data integration and sharing information**.

3 Objective of UCA

The objective of UCA is to provide for **seamless integration** across the utility enterprise using off-the-shelf international standards to reduce costs in several phases of a system life cycle (see Figure 2). 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, 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.

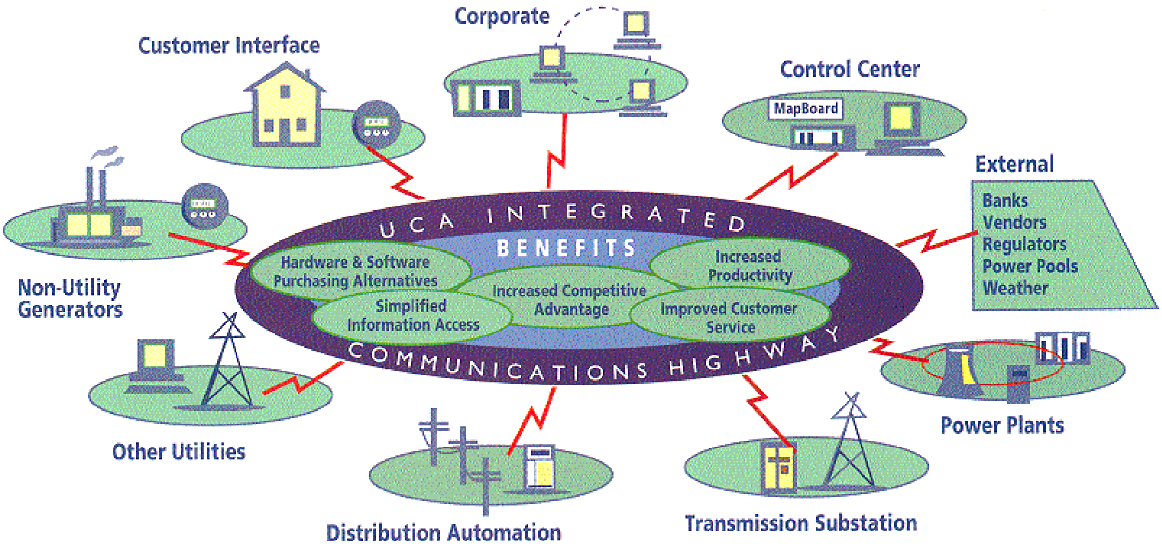


Figure 2 – Seamless information integration based on UCA

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 of the profiles are provided by the standard 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.

4 Utility Communications Architecture (UCA) documents

The UCA documents specify a set of existing international standards which can be applied to specific communications architectural requirements in the utility industry. Information in the documents 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

Modelling 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

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). Several committee drafts of IEC 61850 have already been accepted by IEC TC 57 member countries for final approval in 2001. The relation between UCA and IEC 61850 documents is shown in Figure 3.

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 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.

Plans are underway to set up a European TASE.2 users group in 2001.

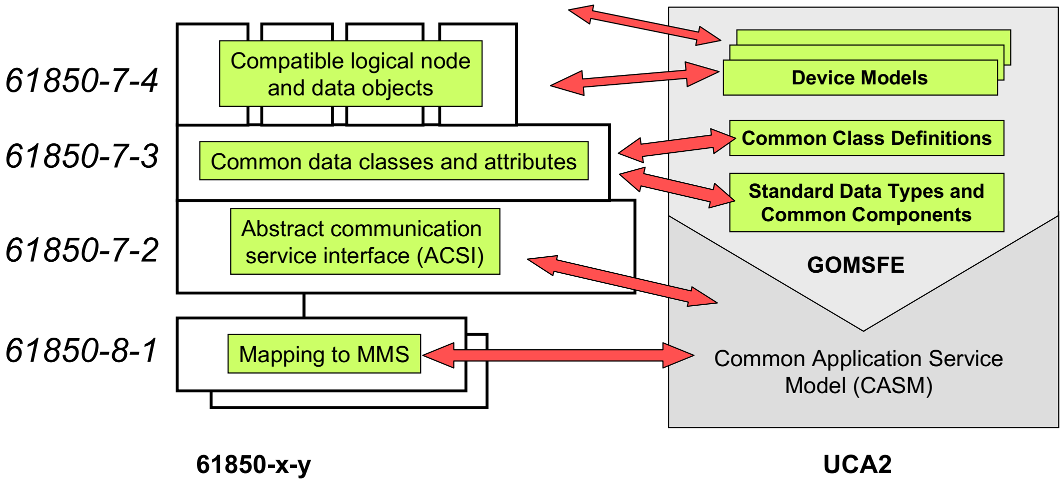


Figure 3 – IEC 61850 and UCA

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 4).

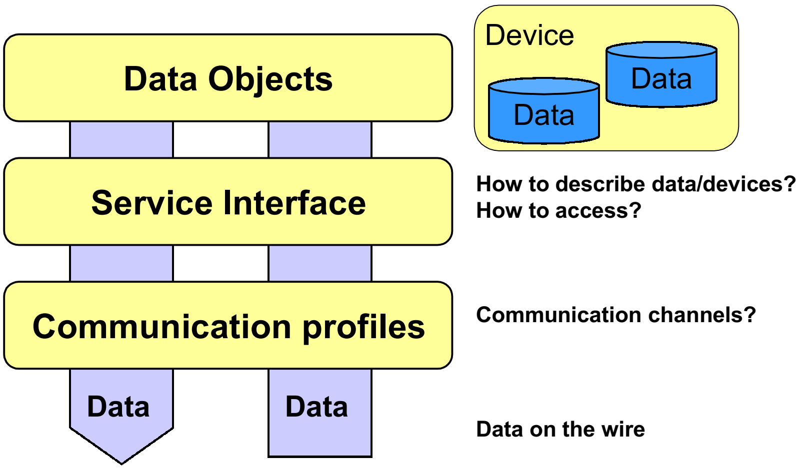


Figure 4 – The three levels of UCA

UCA communications profile

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 and 100 Mbit/s, with 1 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.

Figure 5 lists the complete communications architecture of UCA. For specific requirements a third block shows the reduced stack variants.

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 inter-operability, 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 layer standards. MMS (ISO 9506) is the service specification of choice. Mapping of CASM to MMS is included in the UCA document.

	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	Presentation	CL Pres.	FastByte Pres.					Presentation	Presentation	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			3 Layer				TCP/IP		

Figure 5 – UCA communications architecture

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 23 relays could be modelled using the basic building blocks.

An excerpt of GOMSFE device models are listed below. These models define some 2000 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)
- Synchronizing 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.

5 UCA substation demonstration initiative

EPRI's UCA Substation Communications Automation project 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

In 2001 an UCA users group will be set up to provide many services like conformance testing, training, standardisation, and information dissemination.

6 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.

7 Global TASE.2 adoption

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 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, com-pared 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 150-200 completed or current implementations of TASE.2-compliant systems in the United States (1998).

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.

8 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 fundamental 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,
- Asset management,
- 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 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 6).

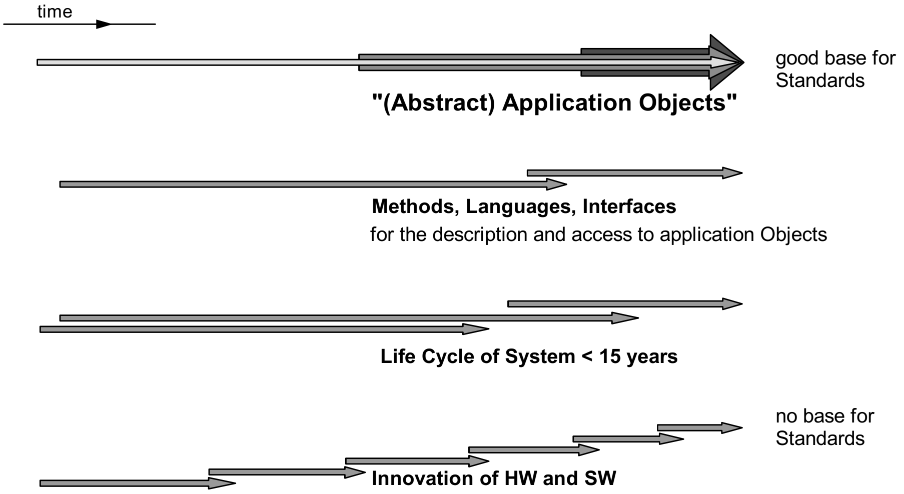


Figure 6 – What is important to be standardised?

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!

9 Benefit of device modelling – a human issue

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, exchanging, and unique interpreting 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 do 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 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 7).

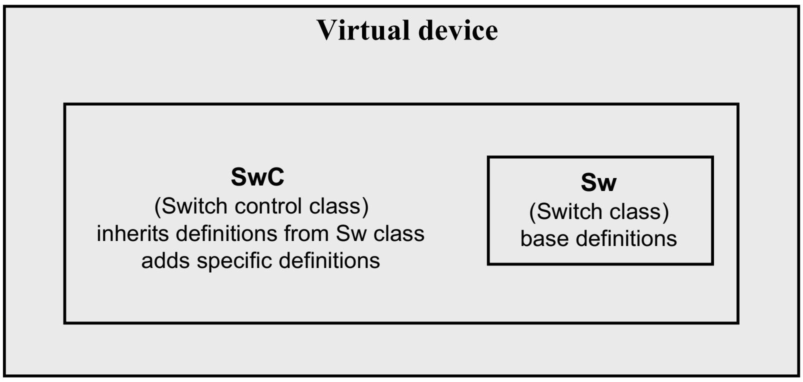


Figure 7 – Switch controller and switch class

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.

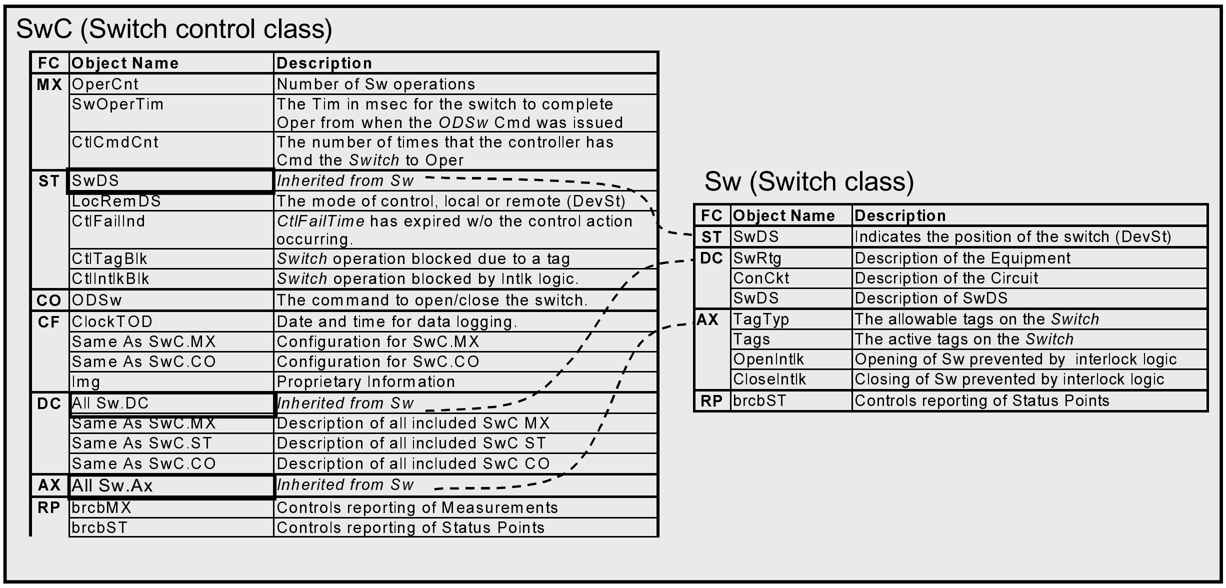
The box in which the switch controller is located is called a “virtual device”. Usually models are abstract in the sense that they do describe only those aspects that are visible to the remote user of a device. It is sufficient to know the external visible data and behaviour of the device (the **WHAT**). 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 8.

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”, and “DC”. These terms (abbreviations) indicate a specific semantic of an attributes. “MX” stands for Measurements, “ST” for status, “CO” for control, “CF” for configuration, and “DC” for description.



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 8 – UCA switch control class

All attributes of the class are named (see 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 that the Data types and the value ranges of the attributes are not shown here, they are defined in the class definitions.

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 all 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 "AccSet" (=Accumulator set").

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, ...

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 have to learn as much system structures (terms, semantics, and syntaxes) as they use different systems.

Assume that we use just 25% of the attributes of IEEE TR 1550 (or IEC 61850) in a substation automation project. That means we use some 750 class attributes hierarchically organised in many classes. What would it cost for a new automation system to specify these from scratch? Much! More expensive than specifying the attributes and an appropriate hierarchy would it be to reach consensus between user and vendor teams of well experienced engineers.

Just a fraction of this total costs 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.

10 Seamless communication

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,...) 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.

High-speed networks usually "copy" cyclically a small fraction of the available information on the network for real-time control. In the future, enterprise applications demand more information accessible at the system, and: reduced integration cost, more advanced technology, more productivity. To meet these objectives, the operators 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.

The vision of a seamless data and device model integration into a utility's information system has at least two aspects. First, the integration of different technologies in one device. Second, the integration of device data in the enterprise system.

Within a device the seamless integration requires according to Figure 9 the mapping of the GOMSFE models to MMS (data dictionary) and additionally, e.g., to a XML document (data dictionary). The first allows **real-time data exchange** within, e.g., a substation. The second can be applied for any **non-critical applications** using standard web browsers. This dual mapping can easily be kept consistent because both mappings use the same model source! The model source, i.e., the GOMSFE models build the source for same mapping.

The utility wide seamless integration applies the very same solutions as within a substation. One difference is that the messages exchanged between systems (MMS and HTTP messages) will be routed throughout the whole enterprise. Another is the increased security requirement when communicating world-wide over wide area networks.

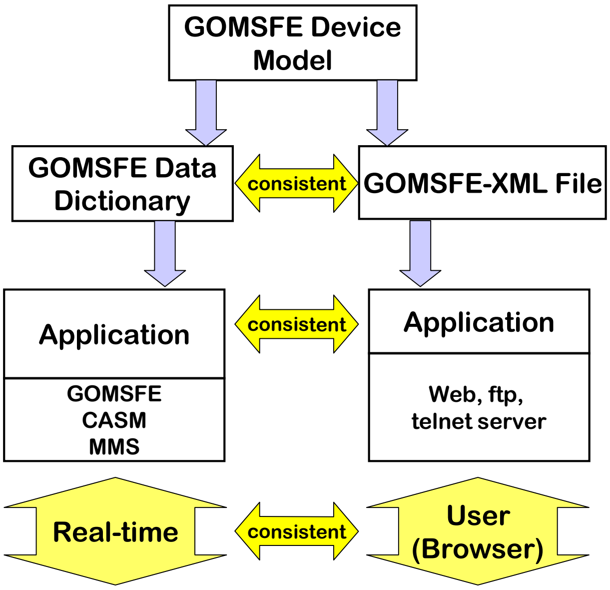


Figure 9 – Seamless and consistent model mapping

UCA and IEC 61850 comprise models, data, services, and communication protocols that provide a utility-wide seamless integration method. Start a UCA client, hook in to the Bitronics PowerServer Meter at IP address 208.176.40.251 and see what is going on in Sterling Heights (Michigan, USA) - it's that easy (required demo software see below).

The implemented models are network-visible. Through appropriate services another device, e.g., a substation supervisory device can retrieve the complete description of the devices data model. These services provide the self-description of the device.

11 Summary

Deregulation will place greater demands for information on utilities than they have experienced before. IEEE's UCA TR 1550 and IEC 61850 provide 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.

The most important issues are the models of the real device data and the rules (service interface) how to access these data. On the other side it is obvious that an appropriate transport mechanism (communication profiles), e.g., the TCP/IP or a point-to-point link, must be used to exchange the messages between devices.

By providing a common communications protocol stack, UCA and 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. The solution provides plant and enterprise wide seamless integration.

A CD ROM available since mid 2000 includes everything needed for a start to learn about UCA/IEC61850/MMS technology (includes also the IEEE TR 1550). 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, Gerhard; Gärtner, W.; Kimpel, T.; Link, V.; März, W.; Schmitz, W.; 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 www.Nettedautomation.com/standardization/IEC_TC57/WG07/etz_report.html

13 Glossary

CASM	Common Application Service Model (Get, Set, Reporting, Logging, Control, ...)
EPRI	Electric Power Research Institute, Palo Alto, CA, USA
GOMSFE	Generic Object Model for Substation Feeder Equipment; Models and meta data of some 100 substation device models (switch, breaker, transformer, RTU, ...).
ICCP	Inter-Control Center Communications Protocol (IEC 60870-6 TASE.2)
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
IEEE	The Institute of Electrical and Electronics Engineers
Metadata	Data about data, e.g., unit of an analog value
MMS	Manufacturing Message Specification (ISO/IEC 9506)
TASE.2	Telecontrol Application Service Element Two
UCA™	Utility Communication Architecture (IEEE TR 1550)

"UCA" is a Trademark of EPRI, Palo Alto, CA, USA