

Energy management system application program interface (EMS-API) —

Part 1: Guidelines and general requirements

The European Standard EN 61970-1:2006 has the status of a
British Standard

ICS 33.200

National foreword

This British Standard is the official English language version of EN 61970-1:2006. It is identical with IEC 61970-1:2005.

The UK participation in its preparation was entrusted to Technical Committee PEL/57, Power systems management and associated information exchange, which has the responsibility to:

- aid enquirers to understand the text;
- present to the responsible international/European committee any enquiries on the interpretation, or proposals for change, and keep UK interests informed;
- monitor related international and European developments and promulgate them in the UK.

A list of organizations represented on this committee can be obtained on request to its secretary.

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the *BSI Catalogue* under the section entitled “International Standards Correspondence Index”, or by using the “Search” facility of the *BSI Electronic Catalogue* or of British Standards Online.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, the EN title page, pages 2 to 41 and a back cover.

The BSI copyright notice displayed in this document indicates when the document was last issued.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 July 2006

© BSI 2006

ISBN 0 580 48967 1

Amendments issued since publication

Amd. No.	Date	Comments

**Energy management system application program interface (EMS-API)
Part 1: Guidelines and general requirements
(IEC 61970-1:2005)**

Interface de programmation d'application
pour système de gestion d'énergie
(EMS-API)
Partie 1: Lignes directrices et exigences
générales
(CEI 61970-1:2005)

Schnittstelle der Anwendungsprotokolle
von Energiieverwaltungssystemen
(EMS-API)
Teil 1: Leitfaden und allgemeine
Anforderungen
(IEC 61970-1:2005)

This European Standard was approved by CENELEC on 2005-12-01. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the Central Secretariat has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Cyprus, the Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland and the United Kingdom.

CENELEC

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

Central Secretariat: rue de Stassart 35, B - 1050 Brussels

Foreword

The text of document 57/777/FDIS, future edition 1 of IEC 61970-1, prepared by IEC TC 57, Power systems management and associated information exchange, was submitted to the IEC-CENELEC parallel vote and was approved by CENELEC as EN 61970-1 on 2005-12-01.

The following dates were fixed:

- latest date by which the EN has to be implemented
at national level by publication of an identical
national standard or by endorsement (dop) 2006-12-01
- latest date by which the national standards conflicting
with the EN have to be withdrawn (dow) 2008-12-01

Annex ZA has been added by CENELEC.

Endorsement notice

The text of the International Standard IEC 61970-1:2005 was approved by CENELEC as a European Standard without any modification.

CONTENTS

INTRODUCTION	5
1 Scope	6
2 Normative References	6
3 Terms and definitions	6
4 System integration.....	7
4.1 Integration scenarios	7
4.2 Integration considerations.....	7
4.3 Component-based interfaces	10
4.4 Relationship to IEC 61968 series of standards.....	11
5 EMS-API reference model	12
5.1 General.....	12
5.2 Control center environment.....	13
5.3 Application context	13
5.4 Application	13
5.5 Component.....	14
5.6 Legacy application and wrappers	14
5.7 Component model	15
5.8 Component container	16
5.9 Component adapter	16
5.10 Component execution system	17
5.11 Middleware.....	17
5.12 Communication profiles	18
5.13 Reference model examples.....	18
6 EMS-API standards	20
6.1 General.....	20
6.2 CIM (IEC 61970-3XX).....	20
6.3 CIS (IEC 61970-4XX)	23
6.4 CIS technology mappings (IEC 61970-5XX)	24
7 General expected infrastructure functionality	24
7.1 General.....	24
7.2 Component Container.....	25
7.3 Middleware.....	26
7.4 Communication Profile Services	26
7.5 Utility-specific services	27
Annex A (informative) Component models.....	28
Annex B (informative) Typical applications and functions	32
Annex C (informative) Utility issues with standard component models	37
Annex D (informative) Examples of component execution systems and middleware products	39
Annex ZA (normative) Normative references to international publications with their corresponding European publications	41
Bibliography	40

Figure 1 – EMS-API Reference Model.....	12
Figure 2 – EMS using EMS-API component standard interfaces	19
Table 1 – Benefits of Component-based Interfaces	11
Table 2 – Examples of EMS application contexts.....	13
Table B.1 – Typical applications and functions	32

INTRODUCTION

This standard is part of the IEC 61970 series that defines application program interfaces (APIs) for an energy management system (EMS). This standard is based to a large extent upon the work of the EPRI Control Center API (CCAPI) research project (RP-3654-1). The principle objectives of the EPRI CCAPI project are to:

- reduce the cost and time needed to add new applications to an EMS or other system¹;
- protect the investment in existing applications that are working effectively;
- improve the capability to exchange information between disparate systems both within and external to the control center environment.

The technical approach is to provide an integration framework for interconnecting existing applications/systems that is

- based on a common architecture and information model;
- independent of the underlying technology.

The principal task of the IEC 61970 series of standards is to develop a set of guidelines and standards to facilitate 1) the integration of applications developed by different suppliers in the control center environment² and 2) the exchange of information to systems external to the control center environment. The scope of these specifications includes other transmission systems as well as distribution and generation systems external to the control center that need to exchange real-time operational data with the control center. Therefore, another related goal of these standards is to enable the integration of existing legacy systems as well as new systems built to conform to these standards in these application domains.

The complete set of standards includes the following parts:

- Part 1: Guidelines and general requirements
- Part 2: Glossary
- Part 3XX: Common Information Model (CIM)
- Part 4XX: Component Interface Specification (CIS)
- Part 5XX: CIS Technology Mappings

IEC 61970-1 provides a set of guidelines and general infrastructure capabilities needed for the application of the EMS-API interface standards. It describes the reference model that provides the framework for the application of the other parts of the EMS-API standards. This reference model is based on a component architecture, which places the focus of the standards on component interfaces for information exchange between applications in a control center environment. The model is also applicable to similar information exchanges between control center applications and systems external to the control center environment, such as other control centers, Independent System Operators (ISOs), Regional Transmission Organizations (RTOs), and Distribution Management Systems (DMS).

IEC 61970-1 also includes general capabilities for the integration infrastructure, which while not part of this standard, is expected to provide certain essential services to support the EMS-API interface standards.

¹ Ideally, an application should be installed on a system with minimal effort and no modification of source code; i.e., the way software packages are installed on a desktop computer. The EMS-API Project goal is to at least approach that ideal by reducing the often significant efforts currently required to install third-party applications in an EMS.

² The control center environment includes traditional transmission control within a utility as well as the newer Independent System Operators (ISOs) and Regional Transmission Operators (RTOs), which are not affiliated with any one utility.

ENERGY MANAGEMENT SYSTEM APPLICATION PROGRAM INTERFACE (EMS-API) –

Part 1: Guidelines and general requirements

1 Scope

This part of the IEC 61970 series provides a set of guidelines and general infrastructure capabilities required for the application of the EMS-API interface standards. This part of the IEC 61970 series describes typical integration scenarios where these standards are to be applied and the types of applications to be integrated. A reference model is defined to provide a framework for the application of the other parts of these EMS-API standards. This reference model is based on a component architecture that places the focus of the standards on component interfaces for information exchange between applications in a control center environment. While the primary objective of the EMS-API is to support the integration of applications within the control center, the reference model is also applicable to information exchanges between control center applications and systems external to the control center environment, such as other control centers, ISOs, RTOs, and distribution systems. This standard describes the role of the other parts of the standard, including the Common Information Model (CIM) in the IEC 61970-3XX series, the Component Interface Specifications (CIS) in the IEC 61970-4XX series, and Technology Mappings in the IEC 61970-5XX series.

This part of the IEC 61970 series also includes general capabilities that are needed by the integration infrastructure to facilitate the exchange of information via the component interfaces specified by the CIS. While the integration infrastructure itself is not part of this standard, it is expected to provide certain essential services to support the EMS-API interface standards. These services are enumerated in Clause 6.

This part of the IEC 61970 series does not specify individual implementations or products, nor does it constrain the representation of information within a computer system application. This standard specifies the externally visible interfaces, including semantics and syntax, required to support the interoperability of products supplied by different vendors.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61970-2, *Energy management system application program interface (EMS-API) – Part 2: Glossary*

IEC 61970-301, *Energy management system application program interface (EMS-API) – Part 301: Common Information Model (CIM) base*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61970-2 apply.

4 System integration

4.1 Integration scenarios

Energy management systems are an assemblage of various software subsystems (SCADA, generation control, load forecast, etc.). The guidelines created for this area (or portions thereof) apply to several different integration scenarios. This work recognizes that existing systems with their unique interfaces will need to evolve to use the interfaces specified in this work. The following different types of integration situations are envisaged. While these are typical scenarios, they are only a subset of the possible examples.

- a) Integration of applications developed by different suppliers into one homogenous system:
In this scenario, independently developed application components (such as an Optimal Power Flow (OPF)) are integrated into a system (which includes supporting infrastructure). This allows the system integrator to more easily integrate an individual component into any system, whether it is an existing system or a newly developed system.
- b) On-line data exchange between separate systems:
The control center needs to communicate with other, separate systems within the corporation and between enterprises. Loosely coupled integration schemes are needed to exchange information. Examples include information exchanges with a DMS, the corporate accounting system, or another EMS. Enterprise integration scenarios generally fit into this category.
- c) Integration of separate systems sharing some engineering data:
This corresponds to the situation where application packages from different vendors use partly overlapping engineering modeling data (such as the impedance of a line segment).
- d) Exchange of serialized data between the same applications in different systems:
Limited integration can be achieved by using file transfer techniques. An agreed upon format is used for this type of export/import exchange. The use of File Transport Protocol (FTP) and an eXtensible Markup Language (XML)-based format file (or document) for the exchange of power system models is an example of this scenario.
- e) Developing a new application in a homogenous system:
This corresponds to vendors or utilities developing new applications for integration into their existing systems (as opposed to integration in other systems) using these standards for the application interface.

4.2 Integration considerations

4.2.1 General

The scope of integration to be supported by these standards falls into two loosely defined categories:

- a) Integration of software components to implement an EMS or similar system.
- b) Integration of independent systems.

To properly interact, both software components and systems need a Common Information Model (CIM) to provide a common, consistent meaning (i.e., semantics) to the information exchanged or accessed. For example, to exchange information about a transformer impedance, the classification of the piece of equipment as a transformer is needed together with the attribute name for the impedance value. With this information, a particular instance of a transformer can be identified with its corresponding impedance. The semantic name of these real world objects (together with their attributes, descriptions, and relationships to other real world objects) is provided by the CIM (see 6.2).

However, in other respects, these two categories of integration as discussed in the following clauses have slightly different needs. The intent of the EMS-API series of standards is to address both, and both are viewed as presenting component interfaces.

4.2.2 Integration of software components into a system

4.2.2.1 General

To integrate independently developed software components into a system, several issues need to be addressed.

a) Software component interactions:

Software components need to interact in a collaborative fashion for the system to function properly. This interaction can be in the form of property access, method invocation, and event handling. The public interfaces consisting of properties, methods, and events shall be identified, and a contract on their use shall be specified in order to support integration of software components. Where similar interaction scenarios exist within the system, the specification of consistent interface patterns simplify system integration and maintenance.

b) Public engineering model data for initialization:

The physical power system and related information is simulated with engineering data as reflected in the CIM. Software components share aspects of this engineering data to perform their functions. Upon software component start-up, a component needs to be initialized with a consistent, accurate model of the real-world system it is simulating. A common interface for access to this public, shared data provides a consistent mechanism for software components to initialize their internal models. Once initialized, software component interaction mechanisms can be used to keep their engineering models up to date.

c) Packaging for deployment:

Software components need to be realized in specific forms for packaging and delivery to system integration. A few major technology frameworks exist in the software industry today, each with their own format for packaging. The specification of standards should promote flexibility in implementation possibilities while facilitating the integration of independently developed software. To accomplish this, software component interactions should be specified in an abstract form that is capable of particular technology and language specializations.

To support this type of integration scenario, the properties, methods, and events used at the component interface need to be standardized as well as the data content of the information exchange.

4.2.2.2 Application categories

The scope of the EMS-API standards includes all applications typically found within a control center environment as well as interfaces to external systems needed to support real-time operations. However, since the intent of the EMS-API standard is to define interface standards rather than to define standard applications, the scope of the project can best be understood by considering the list of application categories that will be supported by the EMS-API standards. The actual packaging of component interfaces specified in the CIS into applications is left to the application suppliers; therefore, any attempt to define a list of individual applications by name would unnecessarily constrain suppliers.

The list of application categories and typical applications supported by this standard can be found in Table B.1. The following is a summary of the list:

- supervisory Control and Data Acquisition (SCADA);
- alarm processing;
- topology processing;
- network applications;
- load management;
- generation control;
- load forecast;
- energy/transmission scheduling;
- accounting settlements;
- maintenance scheduling;
- historical information archival;
- data engineering;
- generic user interface;
- dynamic simulation;
- dispatcher training simulator;
- external systems (e.g., DMS, weather, wholesale power marketing, etc.).

4.2.3 Integration of systems

In the integration of heterogeneous systems, where there is likely to be no similarity in the operating environment nor control over it, the emphasis is more on supporting loosely coupled, asynchronous “document” exchanges. In this context, a “document” may be thought of as a large, rich data structure, such as an XML document. Document exchange implies that the data exchanged is complex, structured, and self-describing. The exchange is more likely to involve individual, atomic information transfers where all information regarding how to handle the information and/or action requested in the transfer is self-contained, rather than multi-step transactions where the handling of the information transfer may be contingent upon previous information transfers or events.

4.3 Component-based interfaces

One goal of the EMS-API standards is to encourage the independent development of reusable software components and facilitate their integration in the construction of control center systems through the development of component interface standards. The software industry, including the major Energy Management System (EMS) vendors and suppliers of application software for an EMS, has undergone an evolution from basing software engineering concepts on top down modular software design to object-oriented approaches to the latest refinement using component-based architectures. The component models embraced by the Common Object Request Broker Architecture (CORBA[®])³, Sun[®]⁴'s Enterprise Java Beans[®] (EJB[®]), and Microsoft[®]⁵'s Distributed Common Object Modeling (DCOM) best exemplify this trend. (See Annex A for a description of these three component models).

These component-based approaches also facilitate the integration of software and complete systems from various sources. For this type of any-to-any integration, XML Web Services provide another Internet-based integration model. XML Web services allow applications to communicate and share data over the Internet using the type of information exchange earlier referred to as "document exchange," regardless of operating system or programming language. These services provide another example of a component execution environment becoming more prevalent in Business-to-Business (B2B) information exchanges. (See Annex A for a description of XML Web Services).

The effect on the EMS-API is to shift the focus to developing standards for software component interfaces for exchanging and accessing public information rather than on standardizing the integration framework services that provide these capabilities. The expectation is that applications that adhere to these standards can then be individually delivered and reused in multiple systems. While other infrastructure services may be needed in an actual system, such as directory services and security, these are not the goal of this standard. In fact, they may be more properly considered the domain of the system integrator or system supplier (i.e., part of the EMS system platform, not a reusable plug-and-play component).

The goal is not to develop standard interfaces for middleware. In fact, the goal is just the opposite – to be independent of any particular set of middleware services. This allows integrators to select the right type and scale of infrastructure for each system. It allows service designs to evolve and innovate, and it simplifies component development as well.

It also means that a software developer does not have to deal directly with these services. The system integrator provides the "glue" to insert these components into the system environment. This gives the integrator more freedom to configure components and choose the services best suited to the needs of the system being implemented (i.e., performance, availability, etc.).

³ CORBA is the trade name of a product supplied by OMG. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

⁴ Sun is shorthand for Sun Microsystems, a USA-based corporation. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the company named.

⁵ Microsoft is shorthand for Microsoft Corporation, a USA-based corporation. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the company named.

The following two examples illustrate this independence of components:

- CORBA components specifically do not require the use of CORBA Notification (or any particular service) as its event system.
- COM+ components are written exactly the same way whether they are deployed in an environment with or without Distributed Transaction Coordinator (DTC) and/or a Microsoft Message Queue (MSMQ).

Table 1 lists some of the benefits of this component-based approach to interface standards.

Table 1 – Benefits of Component-based Interfaces

Explicitly addresses the EMS-API project software plug-and-play goal
Does not prescribe an overall system design nor the choice and design of services
Aligns with overall software industry direction to permit the use of mainstream tools to develop components and configure systems
Frees the EMS-API project from rediscovering and reinventing solutions for the many problems of "plug-and-play" software
Provides a more complete solution including important areas such as packaging (what goes on the component disc CD), documentation, versioning, etc.
Does not require designing and/or prescribing middleware services, particularly event, naming and transaction services, but rather permits the use of commercially-available products to provide these services
Provides a canonical form for the utility specific standards developed on the project, enabling developers to directly machine-translate interface definitions into their preferred interface language: Object Management Group (OMG)/International Standards Organization (ISO) Interface Definition Language (IDL), Java ^{®6} interface syntax or Microsoft IDL
Allows the EMS-API project to focus on designing interfaces and events for utility applications without having to wait for all the middleware problems to be solved. This permits vendors to get to market quicker with EMS-API-compliant interfaces in their application products

4.4 Relationship to IEC 61968 series of standards

The IEC 61968 series of standards deals with system interfaces for distribution management systems. There is a great deal of similarity between these standards and those contained in the IEC 61970 series of standards, not only because of some overlap in scope, but because the IEC 61968 series of standards are also based on the CIM. The IEC 61968 series of standards build on the CIM Base specified in IEC 61970-301 wherever possible by extending it to include additional specializations of existing classes, but also adding entirely new sets of classes to model objects found in the distribution problem domain. Therefore, to comprehend the entire scope of the CIM, it is necessary to review both the IEC 61970 and IEC 61968 series of standards which deal with the CIM.

The IEC 61968 series of standards are also concerned with defining standard information exchanges between distribution business functions, similar to the IEC 61970-4XX standards, but do not attempt to define application program interfaces (i.e., services to be implemented by components). However, the IEC 61968 series of standards envision that the standard messages defined therein can be transferred over the APIs defined in these IEC 61970 series of standards.

⁶ Java is the trade name of a product supplied by Sun Microsystems. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

5 EMS-API reference model

5.1 General

The EMS-API reference model is an abstract architecture that provides a visualization of the problem space being addressed, provides a language for describing and discussing solutions, defines terminology, and provides other similar aids toward achieving a mutual understanding of the problem being solved with the EMS-API standards.

The reference model is not a design, nor is it intended to describe software layers, although a layering approach is hard to avoid and is implied. The primary function is to show clearly which parts of the problem space are the subject of the EMS-API standards, and by contrast, which are outside the domain of the EMS-API project and why. It is also intended to show how different parts of the standard relate to each other.

Figure 1 is a diagram of the reference model, with the shaded areas representing those portions of the reference model that are the subject of this standard. The non-shaded areas represent parts of a system that are essential for creating a framework for reusable application components. Each part of the model is discussed later in this document.

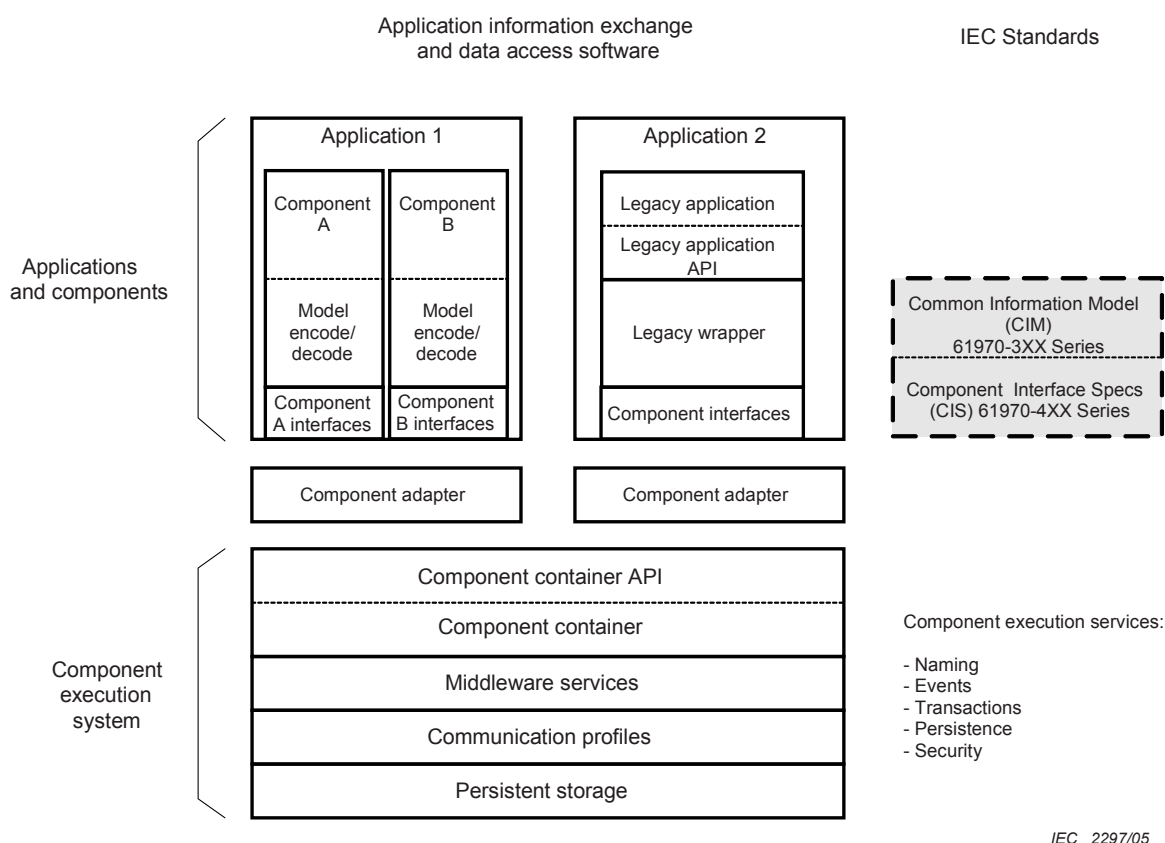


Figure 1 – EMS-API Reference Model