INTERNATIONAL TELECOMMUNICATION UNION

# ITU-T Q.1300

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(10/95)

## INTELLIGENT NETWORK

# TELECOMMUNICATION APPLICATIONS FOR SWITCHES AND COMPUTERS (TASC) – GENERAL OVERVIEW

## ITU-T Recommendation Q.1300

(Previously "CCITT Recommendation")

## FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation Q.1300 was prepared by ITU-T Study Group 11 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 17th of October 1995.

---

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

# CONTENTS

## SUMMARY

This Recommendation gives an overview to the Q.1300-Series of Recommendations which define Telecommunication Applications for Switches and Computers (TASC). The main purpose of TASC is to allow applications running within the network user's environment to integrate telecommunications services with computing facilities. This would typically allow business applications to use TASC to integrate the computer workstation and telephone at the user's desktop. The overview contained in this Recommendation considers the requirements behind TASC and the key concepts leading to its definition. It also defines the vocabulary which is used by the other Recommendations in the TASC series.

**Recommendation Q.1300**

# TELECOMMUNICATION APPLICATIONS FOR SWITCHES AND COMPUTERS (TASC) – GENERAL OVERVIEW

*(Geneva, 1995)*

## 1 Scope

This Recommendation provides a general overview of Telecommunication Applications for Switches and Computers (TASC). The basic concepts, vocabulary and architecture upon which TASC is defined are presented.

## 2 References

The following Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision: all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

- ITU-T Recommendation Q.1301 (1995), *Telecommunication Applications for switches and computers (TASC) – TASC architecture.*
- ITU-T Recommendation Q.1302 (1995), *Telecommunication Applications for switches and computers (TASC) – TASC functional services.*
- ITU-T Recommendation Q.1303 (1995), *Telecommunication Applications for switches and computers (TASC) – TASC management: Architecture, methodology and requirements.*
- CCITT Recommendation X.200 (1988), *Reference model of Open Systems Interconnection for CCITT applications.*
- CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.209 (1988), *Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.217 (1988), *Association control service definition for Open Systems Interconnection for CCITT applications.*
- CCITT Recommendation X.219 (1988), *Remote operations: Model, notation and service definition.*
- CCITT Recommendation X.227 (1988), *Association control protocol specification for Open Systems Interconnection for CCITT applications.*
- CCITT Recommendation X.229 (1988), *Remote operations: Protocol specification.*
- ISO/IEC 9545:1994, *Information technology – Open Systems Interconnection – Application Layer structure (ALS).*

## 3 Terms and definitions

The following vocabulary and terms are used:

### 3.1 TASC

**3.1.1** **client**: A component of an application which invokes a functional service.

**3.1.2** **data terminal**: A device to allow a user to communicate with a computer.

**3.1.3 functional service**: An interaction between a communicating pair of functional entities at their application layer that provides benefit to the entities involved. The functional service is an operation or service available to a client which can be used to invoke a remote operation performed by the server.

**3.1.4 server**: A component of an application which performs a functional service.

**3.1.5 telecommunication device**: A device that allows communication between users such as a telephone or headset.

**3.1.6 first party call control**: The ability for a computer application to manipulate a telecommunication device by acting on that device which is directly connected. The application has access to the same information as the device and can only perform the same functions as the device.

**3.1.7 third party call control**: The ability for a computer application to indirectly manipulate a telecommunication device on a switch via a communication protocol, e.g. TASC. The application can indirectly manipulate a collection of telecommunication devices connected to that switch on behalf of application users. The application has access to information provided by the switch and can request operations to be performed by that switch.

**3.1.8 operation domain**: An Operation Domain defines the set of all possible instances of TASC objects for which the TASC application running on the host computer has visibility via the TASC interface. This includes transitory object instances which cannot be specifically determined during the initial association of the host application with the switch.

**3.1.9 working domain**: A Working Domain is a subset of (or equal to) the Operation Domain. It identifies the set of all instances of TASC objects with which the TASC application wishes to operate (i.e. monitor). This includes transitory object instances which cannot be specifically determined during the initial association of the host application with the switch.

**3.1.10 communication entity (CE)**: A communication entity (e.g. telephone) that originates, terminates or becomes visible in a call.

**3.1.11 communication party (CP)**: An associative object that maintains the relationship between a call and a CE.

**3.1.12 call**: The sequence of actions by the network which results in the end-to-end communications path between end users. A call will be released and all resources returned by act of either the user or the network.

**3.1.13 basic call**: A call involving exactly two CEs.

**3.1.14 call view**: An abstraction of a call which represents the progression of a basic call from the standpoint of a CE involved in the call.

**3.1.15 user**: An entity that makes use of a CE (e.g. initiates or answers a call).

**3.1.16 agent**: A type of user which is registered and distinguished from other users by their ability to sign-on or log-on to systems which coordinate and distribute calls.

**3.1.17 device**: An entity that serves as an endpoint for a call and accepts signalling information from, and provides such information to the Switch. Represented as a CE in TASC.

**3.1.18 event**: A change in the state of a TASC object which is visible to TASC. The application is notified of events by event report messages.

**3.1.19 object**: An intrinsic representation of an entity that is described at an appropriate level of abstraction in terms of its attributes and functions.

**3.1.20 states**: An indication of an object's current condition that permits prediction of the object's future behaviour.

### 3.2 CCITT Recommendation X.200

**3.2.1 application-entity**

**3.2.2 application-process**

**3.2.3 application-service-element**

**3.2.4 open system**

### 3.3 CCITT Recommendation X.217

**3.3.1 application-association; association**

**3.3.2 application context**

**3.3.3 association control service element**

### 3.4 CCITT Recommendation X.219

**3.4.1 invoker-application-entity; invoker**

**3.4.2 performer-application-entity; performer**

**3.4.3 requestor**

**3.4.4 acceptor**

**3.4.5 remote operations**

**3.4.6 remote operation service element**

### 3.5 ISO/IEC 9545

**3.5.1 single association control function**

**3.5.2 single association object**

## 4 Abbreviations

For the purposes of this Recommendation, the following abbreviations are used:

| | |
|---|---|
| ACSE | Association Control Service Element |
| AE | Application Entity |
| AP | Application Process |
| ASE | Application Service Element |
| ASN | Abstract Syntax Notation |
| BER | Basic Encoding Rules |
| CPE | Customer Premises Equipment |
| FS | Functional Service |
| GDMO | Guidelines for the Definition of Managed Objects |
| ISDN | Integrated Services Digital Network |
| ISO | International for Standardization Organization |
| OSI | Open Systems Interconnection |
| PC | Personal Computer |
| ROSE | Remote Operations Service Element |
| SNMP | Simple Network Management Protocol |
| TASC | Telecommunication Applications for Switches and Computers |
| TMN | Telecommunication Management Network |

# 5 Overview

## 5.1 Problem space

TASC is a set of telecommunication services supported by a communication interface between a switch and a computer (see Figure 1). The scope of the services centres around telecommunication calls and entities associated with telecommunication calls such as devices, agents and users. TASC consists of a set of messages which allows a computer to control and have visibility of calls and devices at a switch and a switch to have access to information at a computer. Additional messages provide for agent operations, feature manipulation, supplementary service access, information exchange and management of resources.

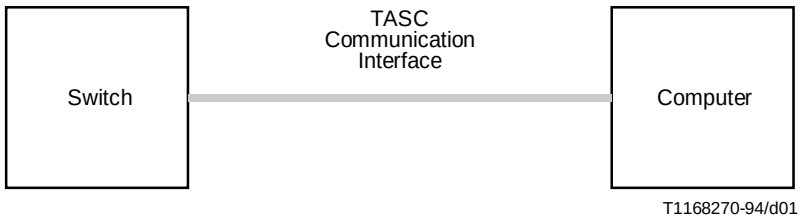


FIGURE 1/Q.1300

**Communication interface**

TASC can be envisioned at a desktop with an agent or other user having telephony capability (e.g. the ability to make and/or receive telephone calls) and data capability (e.g. a terminal screen or personal computer). In a business scenario, information about inbound and outbound calls can be passed to the computer via the TASC interface, the computer can access its database given the information about the call (e.g. calling or called party) received from the switch, and information related to the call and customer can be composed by the computer for presentation to the agent's screen. The TASC control functions can then be used by the computer to coordinate the delivery of the voice and data to the desktop or agent.

By using a combination of call visibility and call control functions, switch-computer applications can be developed to meet the industry needs. The switch provides visibility about calls associated with a telecommunication device. In addition, call control operations can be issued by the computer or user at a data terminal to manipulate inbound and outbound calls.

From a functional perspective, a user at a computer terminal communicates and interacts with an application in the computer (see Figure 2). The computer application in turn communicates with its peer application in the switch to initiate functions for a telecommunications device (e.g. make-a-call, answer-a-call and transfer-a-call). The switch application in turn communicates with the computer application to inform the computer of events happening relative to the telecommunications device (e.g. call-initiated, call-arrived, device alerting, call-establishment). Based on the information received from the switch and the call control functions available to the computer terminal user, the computer can effectively support a wide range of computer applications.

The ability of a computer application to indirectly manipulate a line or device on a switch is referred to as third party call control. This is in contrast to a user directly controlling calls at a telecommunications device via the keypad of the device which is referred to as first party call control. TASC is designed to provide third party call control but can also be used for first party call control.

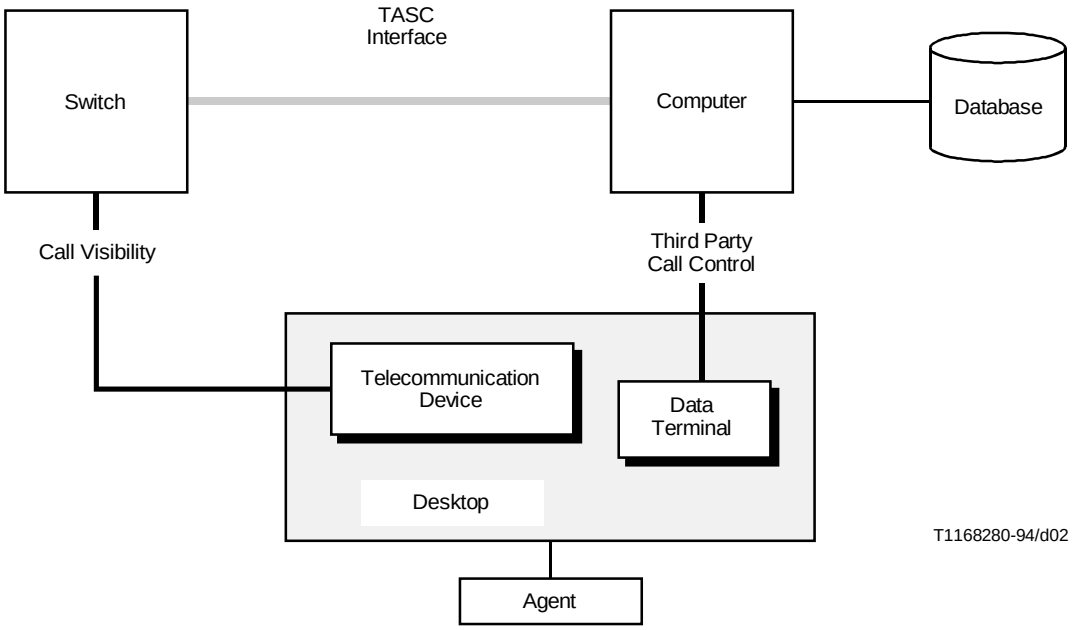


FIGURE 2/Q.1300

**Logical configuration**

The relationship between the computer and the switch is a high-level relationship. The computer issues high-level functions to the switch and the switch then performs low-level operations, which are typically switch dependent, to perform the function. For example, the computer can issue a function to transfer a call and it is then the responsibility of the switch to manipulate the segments or legs of the calls to accomplish the transfer. With TASC, the computer does not attempt to control or manipulate calls on a call segment or call leg basis. This is the province of the switch.

In the same sense, the switch issues high-level functions to the computer. For example, the switch may seek an alternate address to route a call and issue a route request to the computer. It is the function of the computer to find, by whatever means, an alternate address and return it to the switch. The switch does not issue database search and retrieval functions to the computer. This is the province of the computer.

The goal of TASC is to provide a general set of messages which can be used to support a wide range of applications. These applications include:

1) Agent Support;
2) Call Management;
3) Customer Support;
4) Message desk;
5) Emergency Services;
6) Security Services;
7) User Support.

The applications typically involve agents or other users with both telecommunication and computer access. TASC is not dependent upon the equipment configuration and supports a variety of physical configurations (see Figure 3) including:

Type 1 – The case where the end-user has a voice terminal and a data terminal. The data terminal is directly connected to the computer. This configuration allows the end-user to either use the telephone keypad or the terminal keyboard to enter call control functions. In the case where the call control commands are entered using the end-user's data terminal keyboard, the commands received by the computer are formatted into the appropriate TASC message and sent to the switch for processing.

Type 2 – The case where an agent's or operator's voice terminal equipment consists only of a headset. The data terminal is directly connected to the computer. The agent invokes TASC services via the data terminal.

Type 3 – The case is the same as Type 1 except the connection from the data terminal to the computer is accomplished via the switch (e.g. an ISDN basic rate access interface). The ISDN access provides both voice and data connectivity. The function of the data equipment depends on whether it is operating as an intelligent or non-intelligent device:

- Case 1 – For a non-intelligent data terminal, the terminal is logically connected to a computer application over the physical media through the switch (i.e. a terminal connection not shown in Figure 3) and the computer application invokes TASC functions via the interface between the Switch and Computer. The data equipment could also be a PC operating as a non-intelligent data terminal or as an intelligent device communicating with the computer application.

- Case 2 – For intelligent data equipment (e.g. a PC), the data equipment could be directly invoking TASC functions from the switch over the direct connection to the switch. In this case, the TASC link is between the switch and Type 3 PC.

Type 4 – The case where the end-user does not require the use of a data terminal but is handling calls controlled by a TASC application.

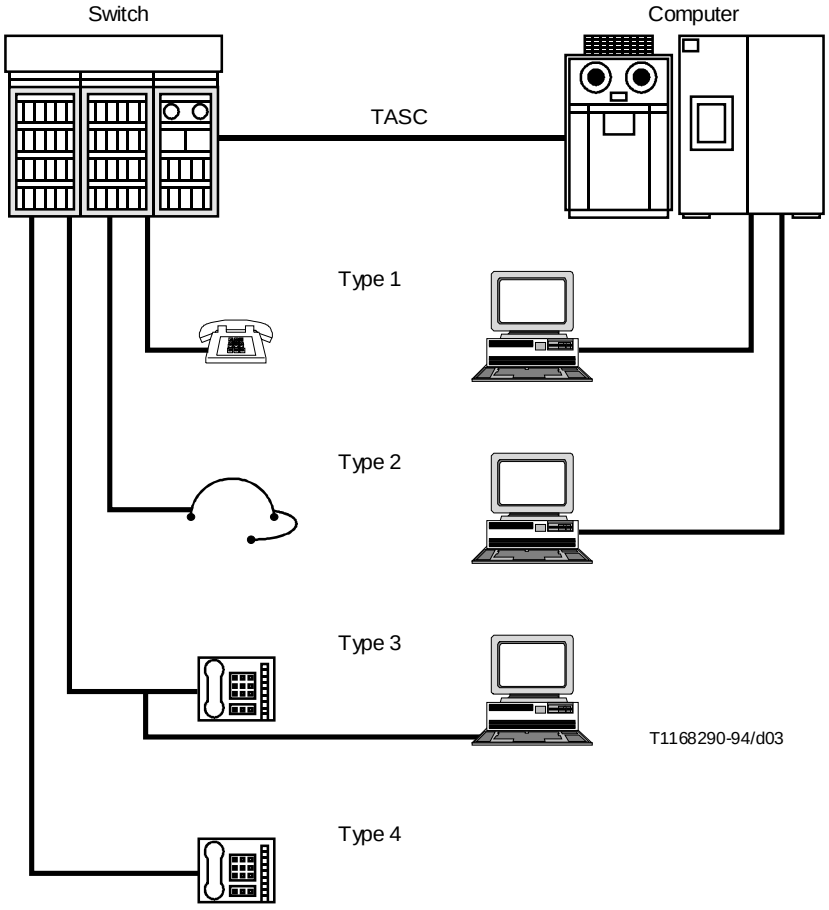


FIGURE 3/Q.1300

**Physical configurations**

TASC can also function in different switching environments on various switching equipment. These environments include transit offices, local exchanges, and private exchanges. Switching equipment includes large and small public and private switches. Computer equipment includes PCs, mainframes and minis. Figure 4 illustrates some possible TASC environments.

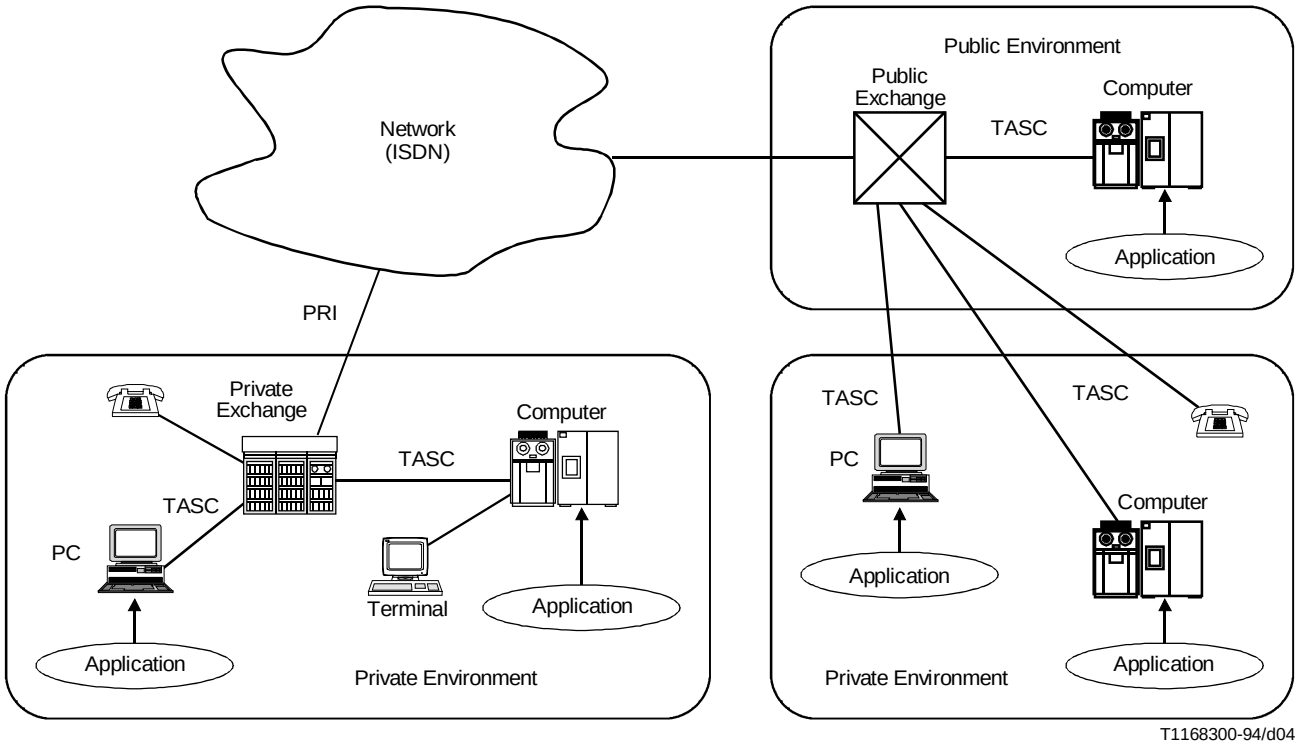


FIGURE 4/Q.1300

**TASC environments**

## 5.2 Messaging

In the OSI environment, communication between Application Processes (APs) is represented in terms of communication between a pair of Application Entities (AEs). Typically, one entity (the invoker) requests that a particular operation be performed and the other entity (the performer) attempts to perform the operation and then report the outcome of the attempt (see Figure 5). For TASC, the invoker of the operation is the Client and the performer of the operation is the Server. For any given operation, the switch and the computer can take on the role of the client or the server.

The messaging between the AEs takes on the form of a Request message issued by the client AE to the server AE to perform an operation, and a Response message issued by the server AE to the client AE indicating outcome of the attempt. The Request-Response message combination is referred to as an operation. These operations may be classified according to whether the performer of an operation is expected to report its outcome. These operations may also be classified according to two possible modes: synchronous, in which the invoker requires a reply from the performer before invoking another operation; and asynchronous, in which the invoker may continue to invoke further operations while awaiting a reply. TASC supports the following types of operations[1)]:

- Operation Class 1: synchronous, reporting success or failure (result or error).
- Operation Class 2: asynchronous, reporting success or failure (result or error).
- Operation Class 3: asynchronous, reporting failure (error) only, if any.
- Operation Class 4: asynchronous, reporting success (result) only.
- Operation Class 5: asynchronous, outcome not reported.

1) Refer to Recommendation X.219 Remote Operations.

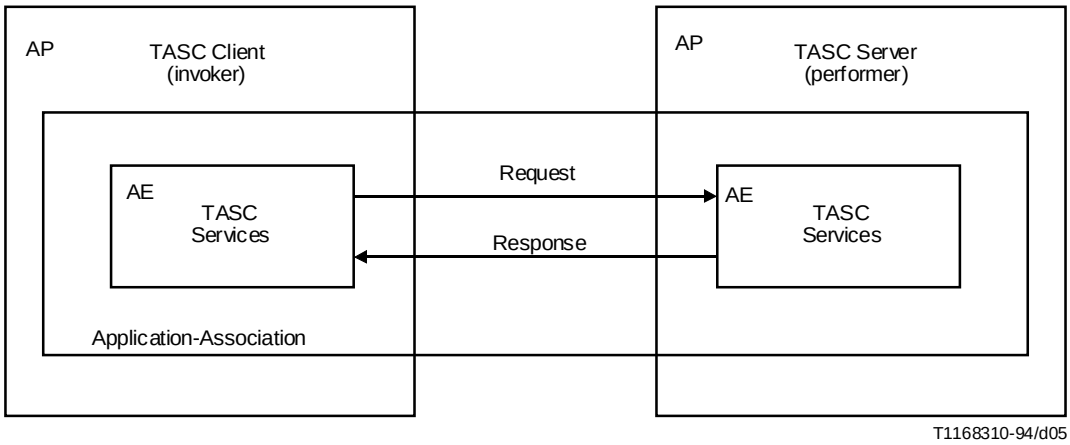


FIGURE 5/Q.1300

**Messaging**

The application-association defines the relationship between a pair of AEs and is formed by the exchange of application-protocol-control-information. The AE that initiates an application-association is called the association-initiating AE or the association-initiator, while the AE that responds to the initiation of an application-association by another AE is called the association-responding AE or the association-responder. Only the association-initiating AE may release an established application-association. Application-associations are classified by which application-entity is allowed to invoke operations:

- Association Class 1: Only the association-initiating application entity can invoke operations.
- Association Class 2: Only the association-responding application entity can invoke operations.
- Association Class 3: Both the association-initiating and the association-responding application entities can invoke operations.

## 5.3 Functional service

The TASC Functional Service (FS) is used by a TASC Client to communicate with a TASC Server. The TASC FS defines an operation to be performed by the server. The FS may or may not have a response from the server. TASC does not specify how FSs are implemented but does define the operation to be performed, resulting events which may be reported, and the associated data. See Figure 6.

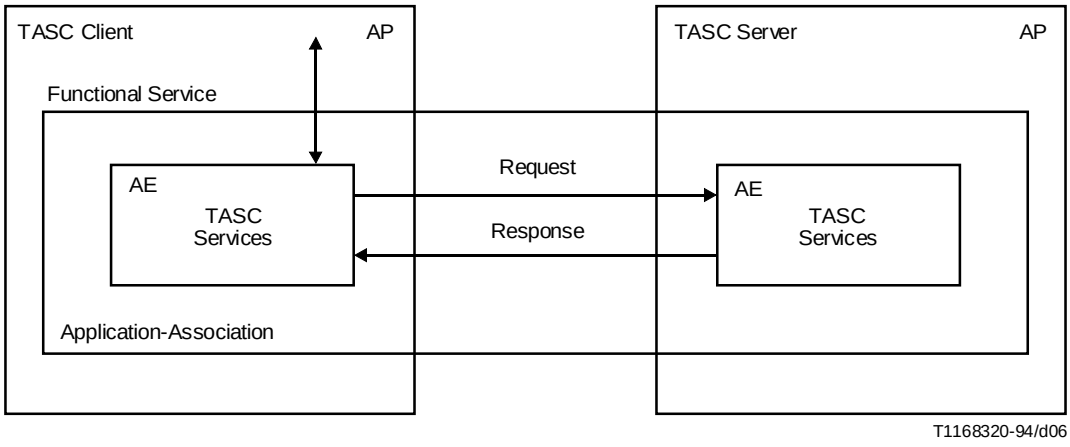


FIGURE 6/Q.1300

**Functional service**

## 5.4 Interface architecture

TASC is based on the International Organization for Standardization (ISO) seven layer Open System Interface (OSI) architecture. At layer seven, the application layer, two peer applications, one in the switch and one in the computer, communicate via a set of functions or operations over a logical communication path called an association. The TASC functional services are conveyed via the TASC communications interface.

Since TASC resides at the application layer, it is independent of the underlying transport method. The operations can be carried over a full OSI stack which includes Presentation, Session and Transport layers or the operations can be carried over smaller stacks such as X.25 and ISDN which map onto the application layer via convergence modules. Thus, TASC is transport independent and can be supported by many different transport protocols and physical media (see Figure 7).

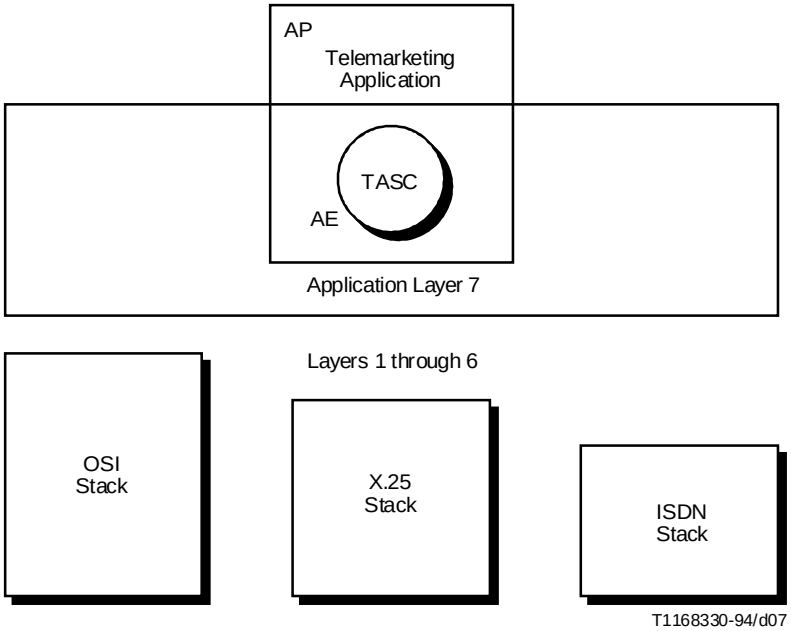


FIGURE 7/Q.1300

**Possible communication arrangements**

## 5.5 Message encoding

The FS definitions may be used to develop the TASC protocol. It is envisioned that this will be done in terms of ROSE operations using ISO's Abstract Syntax Notation One (ASN.1). The standard specifies that the syntax be encoded using ISO's Basic Encoding Rules (BER).

## 5.6 Functional services and objects

TASC uses object-oriented methodology to identify and define objects. The TASC functional services act upon these objects or convey information about these objects or their behaviour. The objects are defined in abstract terms independently of implementation. In general terms, the objects of interest to TASC include:

1) Call;

2) Telecommunication Device;

3) Users and Agent.

A call is abstracted as a Call View (CV) object and a Telecommunication Device is abstracted as a Communication Entity (CE). Communication entities include lines and distribution entities.

Other objects are also defined to support specific applications such as Automatic Call Distribution. It should be noted that TASC does not define all possible switch, computer or data terminal objects since many of these objects are outside the scope of TASC and are not directly visible or manipulatable via TASC functional services.

Figure 8 illustrates a scenario containing some TASC objects.

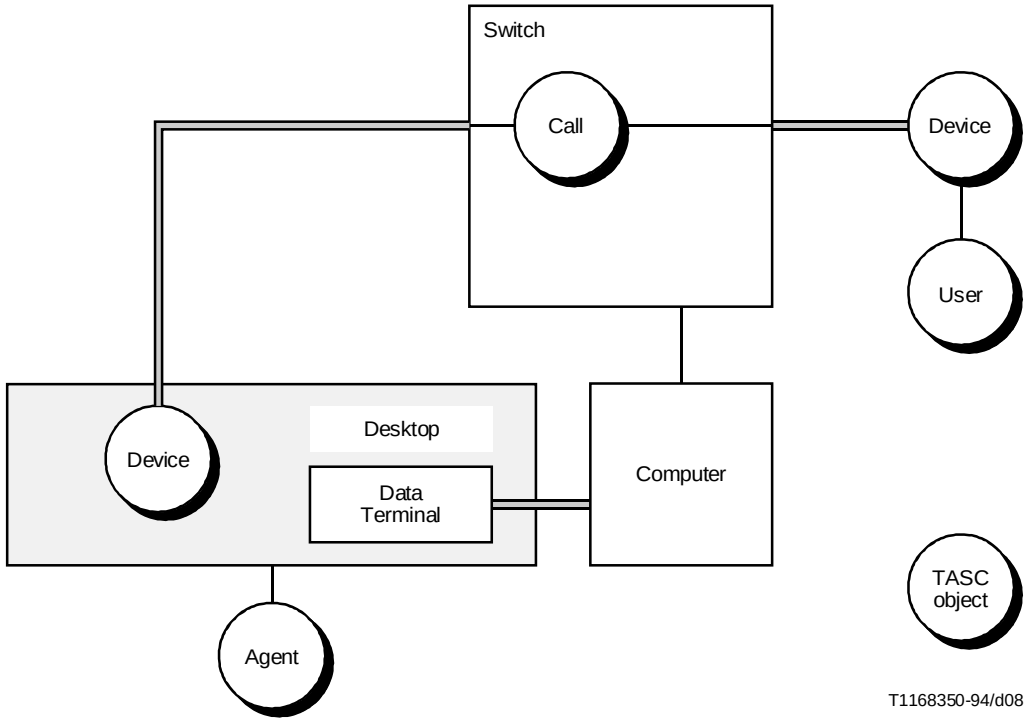


FIGURE 8/Q.1300

**Objects**

## 5.7 Monitoring

Monitoring is the function of reporting upon the behaviour of objects. By placing a monitor on an object, the object can be observed via events reported by the switch.

Monitoring of objects can be accomplished in two ways:

1) statically before association, via subscription arrangements, which indicate which devices are to be visible and which events are to be reported for each device; or

2) dynamically via a monitor function issued by the computer to the switch which indicates which objects are to be visible and the events to be reported for each object.

## 5.8 Management

In order for the TASC interface between a switch and computer host to operate efficiently and reliably, it is necessary to provide a management activity. Whilst management of each end of the interface could be carried out by proprietory means, this would generally require a high degree of manual orchestration. The purpose of TASC Management is to describe a management interface between the switch and computer environment so as to assist in automating the management orchestration.

It considers how to orchestrate the management functions in the switch and computer application support environments. It does not consider how the management functions in the switch and computer application support environment work.

Management functions within TASC will need to interwork with management functions for switching. As TMN is expected to provide architecture for managing some switch networks, then TASC management functions will need to interoperate in a manner compatible with TMN. This will involve the use of managed objects for communication and identification of the appropriate TMN interface.

Fundamental to TASC are the two partners: the computing and switch environments. Each has its own management aspect which must be functionally coordinated in order to provide TASC management:

- The Computer Management function, associated with the TASC application running on the host computer.
- The Switch Management function, associated with the telecommunications domain.

This arrangement is represented by Figure 9, where the computing and switch environments are portrayed. Each environment has its own management function which communicate together to make up TASC Management. In addition, the management function for each communicates with the "application" (TASC or Switch) which operates in that environment.

TASC Management information will be conveyed at the reference point between the management functions for the computer and switch environments.

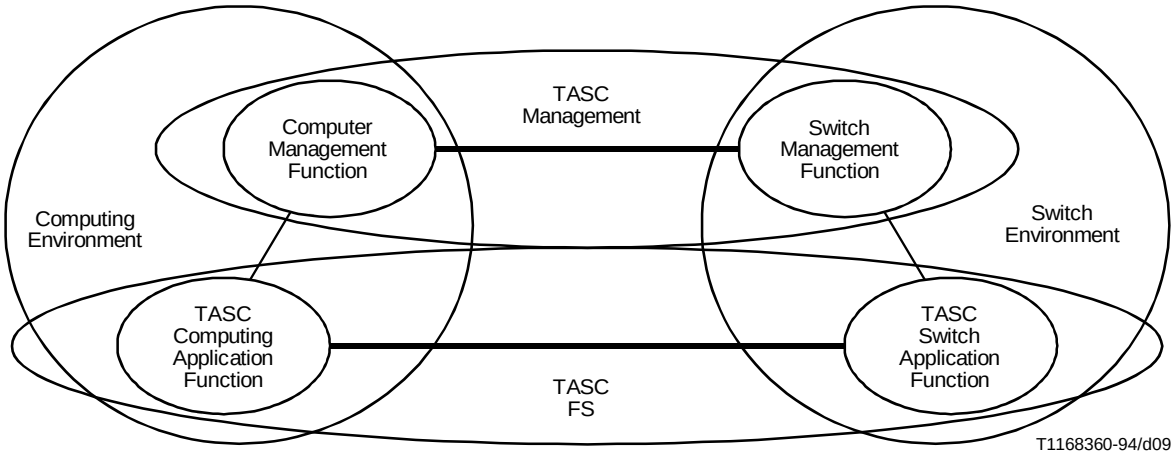


FIGURE 9/Q.1300

**Management architecture**

Both GDMO, the basis for TMN object definitions, and SNMP representations should be supported to allow the corresponding interface mechanisms to be used with TASC. This may be achieved by restructuring GDMO definitions to fit into an SNMP framework. Thus, GDMO would become the base source.

## 6 Structure of Recommendation

The Q.1300-Series Recommendations is organized into the following Recommendations:

– ITU-T Recommendation Q.1300 – TASC: General Overview.

– ITU-T Recommendation Q.1301 – TASC: Architecture.

– ITU-T Recommendation Q.1302 – TASC: Functional Services.

– ITU-T Recommendation Q.1303 – TASC: Management.

The Architecture Recommendation describes the objects and their models.

The Functional Services Recommendation describes the semantics of the TASC functional services.

The Management Recommendation provides a general overview of the management model, management principles and management architecture.

# Appendix I

## Application context guidance

(This appendix does not form an integral part of this Recommendation)

TASC may utilize ITU-T Recommendations to define modules, application service elements, and application contexts. The abstract syntax may be used to define the application service element. The application service element can then be used to define the application context. ISO compliant Object Identifiers should be assigned to the modules, application service element, and application context. These TASC Object Identifiers are used during association establishment to define the abstract context (i.e. the protocol) to be used for communication between two peer applications. See Figure I.1.

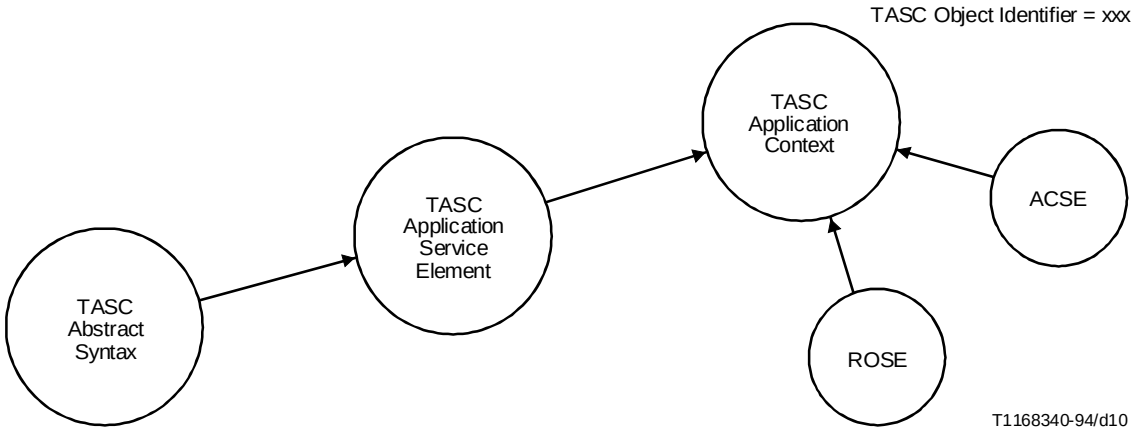


FIGURE I.1/Q.1300

**TASC application context**