INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Z.361
(02/99)

SERIES Z: LANGUAGES AND GENERAL SOFTWARE ASPECTS FOR TELECOMMUNICATION SYSTEMS

Man-machine language – Specification of the man-machine interface

---

Design guidelines for Human-Computer Interfaces (HCI) for the management of telecommunications networks

ITU-T Recommendation Z.361

(Previously CCITT Recommendation)

ITU-T Z-SERIES RECOMMENDATIONS

LANGUAGES AND GENERAL SOFTWARE ASPECTS FOR TELECOMMUNICATION SYSTEMS

FORMAL DESCRIPTION TECHNIQUES (FDT)	
Specification and Description Language (SDL)	Z.100–Z.109
Application of Formal Description Techniques	Z.110–Z.119
Message Sequence Chart	Z.120–Z.129
PROGRAMMING LANGUAGES	
CHILL: The ITU-T high level language	Z.200–Z.209
MAN-MACHINE LANGUAGE	
General principles	Z.300–Z.309
Basic syntax and dialogue procedures	Z.310–Z.319
Extended MML for visual display terminals	Z.320–Z.329
Specification of the man-machine interface	**Z.330–Z.399**
QUALITY OF TELECOMMUNICATION SOFTWARE	Z.400–Z.499
METHODS FOR VALIDATION AND TESTING	Z.500–Z.599

For further details, please refer to ITU-T List of Recommendations.

ITU-T RECOMMENDATION Z.361

DESIGN GUIDELINES FOR HUMAN-COMPUTER INTERFACES (HCI) FOR THE MANAGEMENT OF TELECOMMUNICATIONS NETWORKS

Summary

The design of network management user interfaces is critical for the successful operation of telecommunications networks. This Recommendation provides a set of guidelines for the design of Human-Computer Interfaces (HCI) for the management of telecommunications networks. The general body of human factors knowledge and practice is important to these designs. In addition, telecommunications network management user interfaces can benefit from the application of the additional design guidelines in this Recommendation.

Annex A contains a number of important generic design principles selected from the HCI literature.

Appendix I contains additional reference material.

Appendix II contains a technical framework which relates these guidelines to the HCI reference model provided in Recommendation Z.352

Source

ITU-T Recommendation Z.361 was prepared by ITU-T Study Group 10 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on the 12th of February 1999.

Keywords

Design guidelines, Human-Computer Interface (HCI), Network Management, OAM&P, Telecommunications Management Network (TMN), User Interface.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the 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.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation the term *recognized operating agency (ROA)* includes any individual, company, corporation or governmental organization that operates a public correspondence service. The terms *Administration, ROA* and *public correspondence* are defined in the *Constitution of the ITU (Geneva, 1992)*.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

Page

Introduction

A user-centered design approach, including vital usability testing, is appropriate and recommended for the design of the Human-Computer Interfaces (HCI) for telecommunications network management applications. These user interfaces are in the area of operations, administration, maintenance and provisioning (OAM&P) of telecommunications networks. The guidelines in this Recommendation are provided to enhance the efficiency, usability and effectiveness of network management user interfaces by making them

- easier to learn and remember;
- less likely to produce errors;
- more efficient in accomplishing goals;
- more pleasing to use.

Telecommunications network management applications have specific characteristics which should be addressed when designing the user interface. Collectively, the following aspects differentiate these applications.

- *Telecommunications network management HMIs can be time critical* – In the management of a telecommunications network, and specifically the fault management application, faults may be critical and demand attention immediately. Thus the notions of asynchronous interrupts and the need to access specific information coupled with specific actions are vital for the user.
- *Errors can cause great damage to the network* – The potential to disrupt significant sections of large telecommunications networks carrying large volumes of traffic demands a rigorous approach to the means of ensuring that problems or troubles do not go undetected.
- *Users handle large volumes of data, often in real time* – Network control centres can be likened to traffic control centres where large numbers of apparently isolated events may occur. Providing the user with assistance in prioritizing, filtering and managing these inputs is essential in the design of effective network control applications.
- *Telecommunications networks are often large complex systems* – In a typical office application, a user is confronted with files, documents, and possibly spreadsheets. A typical network control application deals with complex central office switches, optical and other broadband transport equipment, facilities both owned and leased, plus a host of other equipment types each of which has different capabilities and requirements.

Recommendation Z.361

DESIGN GUIDELINES FOR HUMAN-COMPUTER INTERFACES (HCI) FOR THE MANAGEMENT OF TELECOMMUNICATIONS NETWORKS

(Geneva, 1999)

1 Scope

This Recommendation provides guidelines for the design of the G interface of a Telecommunications Management Network (TMN). The G interface is the human-to-workstation interface of a TMN. The reference model for a TMN is defined in Recommendation M.3010.

2 References

The following ITU-T 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 M.3010 (1996), *Principles for a telecommunications management network.*

– ITU-T Recommendation M.3100 (1995), *Generic network information model.*

– CCITT Recommendations Z.301-Z.341 (1988), *Man-Machine Language (MML).*

– ITU-T Recommendation Z.351 (1993), *Data oriented human-machine interface specification technique – Introduction.*

– ITU-T Recommendation Z.352 (1993), *Data oriented human-machine interface specification technique – Scope, approach and reference model.*

3 Terms and definitions

This Recommendation defines the following terms:

3.1 Human-Computer Interface (HCI): The HCI is an implementation of the G reference point of the TMN.

3.2 Human-Machine Interface (HMI): HMI is the term used in several Z-series Recommendations and refers to the same interface as the newer term, HCI.

3.3 Telecommunications Management Network (TMN): A TMN is specified by the M.3000 series of Recommendations.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations:

HCI Human-Computer Interface

HMI Human-Machine Interface

OAM&P Operations, Administration, Maintenance and Planning

TMN Telecommunications Management Network

5 Conventions

As this Recommendation deals in large part with graphical user interfaces, it will present graphic examples where practical. These examples are not intended to imply design solutions, nor are they intended to imply a particular toolkit. Therefore, the figures in clause 6 are provided for illustrative purposes only and their content is not mandatory.

6 Design guidelines

The application of one of these design guidelines used in isolation from other concerns or other guidelines can result in less than optimal or even inappropriate solutions. Therefore, the following guidelines should be applied in a balanced manner along with the designer's understanding of good human factors practice.

6.1 Avoiding the presentation of unnecessary information

6.1.1 Background

The user interface will present information to the user, depending on the application and the current state of that application. This guideline addresses the degree to which peripheral information is included in presentations, in particular for network surveillance tasks.

6.1.2 Design guideline

The amount of incidental or peripheral information presented to the user should be minimized. The information that is presented should be that which is required and appropriate to:

a) perform the required actions;

b) detect important signals or changes in the objects;

c) understand the context of the current displayed objects.

6.1.3 Rationale

Graphic displays should communicate their current state to the user clearly, unambiguously and directly. Users should be able to detect all relevant information, should not misplace or lose important information or functions, and should not lose their way around the interface. This is especially true when network problems or emergencies occur.

There are two basic contributors to human ability to detect signals in any environment: the strength and quality of the signal, and the noise or irrelevant data in which signals are always embedded. Noise impacts all perceptual dimensions and can affect signal interpretation.

The effort required to work in noisy environments also has negative physical and emotional effects. For example, displays with many and/or bright colours and containing too much unnecessary information will cause visual fatigue, irritations of the eyes, muscle tension, and may lead to chronic headache and other stress-related complaints.

6.1.4 Examples of design implications

a) In general, there should be no more than three levels of information within a given display:

   1) Background (the level containing the frame of reference or context, e.g. geographical maps or schematic diagrams);

   2) Middleground (the level which contains the objects of interest to the user, e.g. nodes in the network);

   3) Foreground (the level which contains the most important signals to the user, e.g. display elements used to differentiate alarmed objects in the network).

   These levels should be differentiated in terms of brightness and/or saturation variations, with the highest brightness and/or saturation given to the foreground.

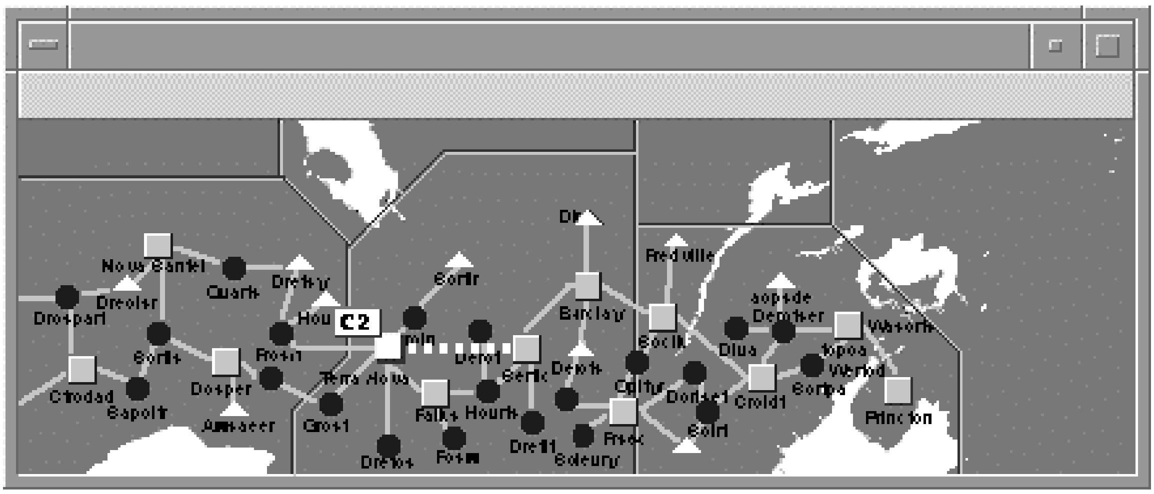
b) Displays should limit the number of colours that are presented simultaneously in the foreground. As a rule of thumb, four different hues could be displayed simultaneously without overloading the display.

c) Use simple graphical shapes. Avoid overuse of highlights and excessive detail.

d) Simplify complex networks through the use of containers for presentation when large numbers of nodes are involved. Containers are objects which hold instances of other objects.

6.1.5 Graphic examples

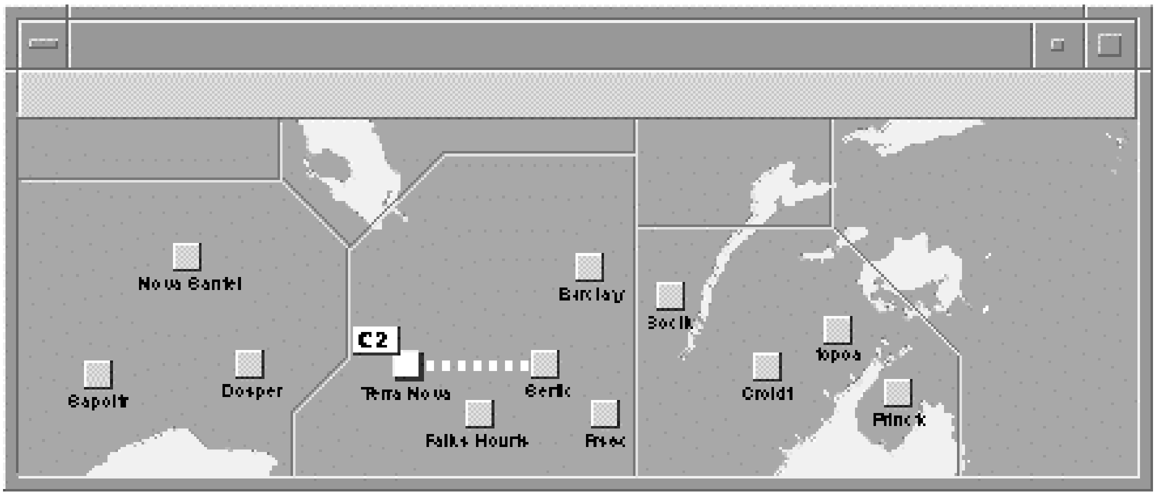
Figure 6.1-1 is an example of a noisy display. The designer has chosen to show all possible nodes and links. Different shapes are used to designate some characteristic of the nodes, and saturated colours and strong backgrounds are present. Detecting new alarm information is critical to some tasks and is not made easier by the excess information in this display.



T1010880-98

Figure 6.1-1/Z.361 – A cluttered display

The example in Figure 6.1-2 demonstrates a minimal background, with a limited set of major nodes displayed for reference. The ability of the user to detect changes or important signals is greatly enhanced. The presence of the alarm (C2) is clearly apparent to the user.



T1010890-98

Figure 6.1-2/Z.361 – Example of minimized information

6.2 Maintaining appropriate context

6.2.1 Background

Network management user interfaces generally deal with tasks such as surveillance by presenting various views of the physical or logical network. Windows are often opened in order to enable the user to accomplish specific tasks, for example to manipulate the state of a particular piece of network equipment. Some designs do not adequately link the subsequent window or task manager screen to the previous operational context. As the tasks may be complex or involve a number of screens, it becomes increasingly difficult for users to remember the original context. Users need to be able to maintain the relevant context from which applications are accessed, worked on and eventually closed.

6.2.2 Design guideline

The context from which specific task managers or applications are launched should be kept visible and accessible until the task is completed or the application closed.

6.2.3 Rationale

Interfaces have been designed in which the user is obliged to open a large number of windows in order to complete a task. These windows may mask or cover up the high-level context of the application, thus preventing the user from being able to see or recover the original network map or view. Complex tasks requiring problem-solving should be facilitated by keeping visible the relationship between the application and the relevant context.

6.2.4 Examples of design implications

a) The default window size and placement should be set so that, when opened, it does not completely cover the previous view.

b) The application to which windows belong should be shown in the window headers.

c) Critical information such as alarm banners should never be covered by the application dialogue windows.

d) Screen pictures or windows should contain information that clearly indicates the source from which they were generated or derived.

6.2.5 Graphic example

In Figure 6.2-1, the detailed view and the overall topographical map both indicate the same network region and the fact that there is an alarm present. The user can easily recognize what region the affected element belongs to. This is an example of geographic context.

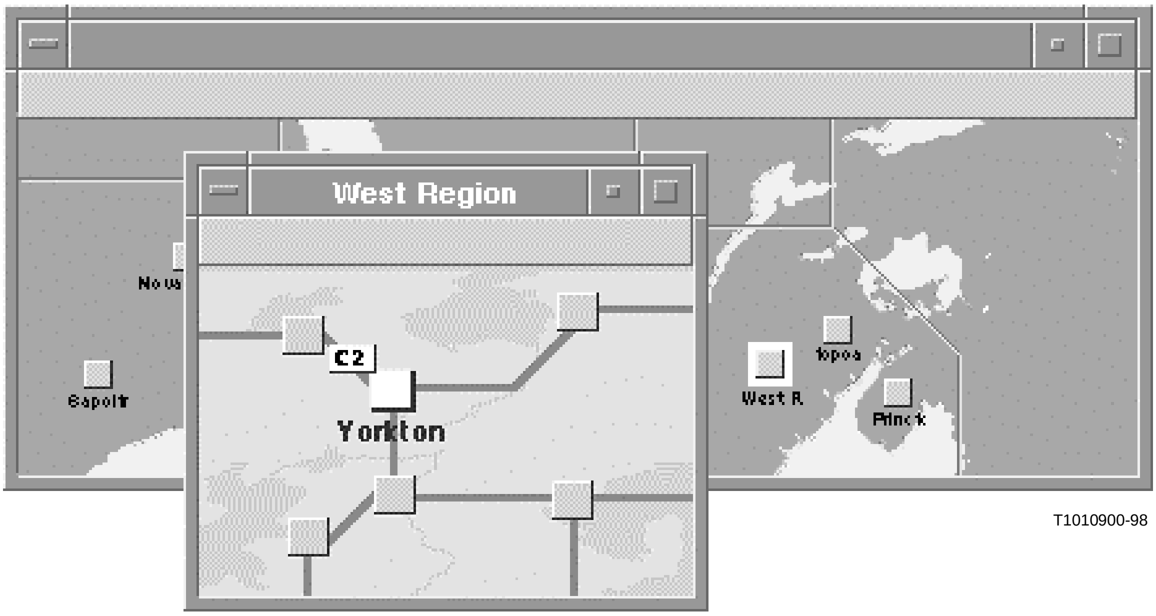


Figure 6.2-1/Z.361 – Maintenance of context

6.3 Providing redundant coding in all important graphic displays

6.3.1 Background

Network management applications are often required to present information to users which requires them to take action. In fault management, for example, alarms may require immediate action if network degradation or failure is to be avoided. This guideline is a more general case of the guideline in 2.1.3/Z.323 item a), which points out that colour should not be used as the only visual coding mechanism.

6.3.2 Design guideline

When presenting time-critical or important information about HCI objects, a minimum of two visual cue dimensions (e.g. shape, size, colour, position, etc.) should be varied simultaneously. Such time-critical information should be presented in the foreground as defined in guideline 6.1.

6.3.3 Rationale

A given network element may have any number of active alarms, some of them acknowledged, others not. In addition, the network element may be in one or more states. Communicating this complexity to the user by varying only one dimension will result in complex and extensive coding schemes which will be very difficult to interpret and remember. It is easier to tell one object from another if their colours, shapes, or other attributes are different than if they were identical in all but one of these dimensions. This multidimensional perceptual presentation allows people to overcome physical limitations such as colour blindness, and environmental adversities such as poor lighting.

6.3.4 Examples of design implications

a) Colour should not be used on its own as a mechanism to communicate important or time-critical changes of state. Colour can enhance the effectiveness and attractiveness of carefully-designed displays. However, note the following:

   1) A significant proportion of the population is partially colour-blind. Of this population, the largest proportion has difficulty in discriminating between red and green.

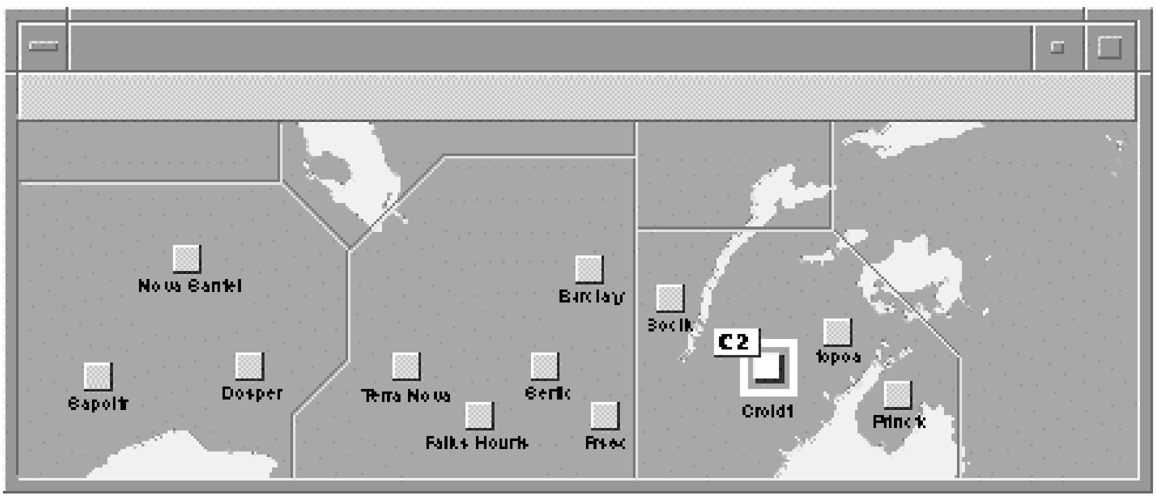
   2) Colour displays are notoriously difficult to control and calibrate properly. Improper settings not only cause stress and physical discomfort, but can also mask information that is coded in colour.

   3) Displays should effectively convey important information to the user even if the display mechanism is degraded.

b) It is desirable that one of the dimensions included in the coding mechanism for display of object state be textual. Text information, used intelligently in conjunction with other graphic displays, can enhance the effectiveness of the display by providing an explicit verbal label of the state for the user. This will help users communicate among themselves, and will facilitate access to support documentation.

6.3.5 Graphic example

The example in Figure 6.3-1 shows the situation when an alarm has been raised against a particular node. Redundant coding is realized by changing the node's shape, adding some critical information and by changing the colour of the node.



T1010910-98

Figure 6.3-1/Z.361 – Redundant coding

6.4 Flattening the hierarchy

6.4.1 Background

Most people have problems when traversing hierarchical user interface systems because they often lose their context or frame of reference, becoming lost in the hierarchy. The extra steps required to traverse the hierarchy adds both time and effort to the task, increases the cognitive load, and decreases the user's satisfaction with the product.

6.4.2 Design guideline

The user interface structure should minimize the number of steps required for a given task. The interface should not force users to traverse the system's hierarchies.

6.4.3 Rationale

Complex systems are typically hierarchically organized and implemented. Often this hierarchy is reflected in the user interface of the systems via hierarchical menus and modes. For example, in dealing with an alarm, users often are given an indication, graphical or otherwise, that an object is alarmed somewhere in the hierarchy. The user is then required to:

a) traverse the network of data until the object is located;

b) determine the state of the object; and

c) act upon the situation.

Also, users are often required to open a number of separate windows in order to find required information for task resolution. Besides slowing down the analysis and understanding of the problem, users may also get lost when a large number of windows is open for each task.

The hierarchy may be required for implementation purposes, but should not be forced on the user through the user interface when an appropriate collection of information and controls can make accomplishing the task more efficient and effective.

6.4.4 Examples of design implications

a) All of the primary information required to understand the context, scope and significance of the situation should be directly available to the user within the current display.

b) Supplementary information needed in a given situation should be made available to the user within one step of the current display.

c) The user should have direct access to all of the primary functionality required to resolve a given situation directly within the current display.

d) Supplementary functionality needed to resolve a given situation should be made available to the user within one step of the current display.

e) Providing direct access to information required by users should not prohibit other means of searching through the data.

f) In some cases, relevant information may be presented directly without requiring any user action. An example of this is when an alarm is displayed, the name of the network element and the severity of the alarm can be presented directly to the screen, without the user having to ask for the information.

6.4.5 Graphic examples

The example in Figure 6.4-1 shows the situation where the user has had to open a large number of windows in order to assemble the information required to accomplish a task.

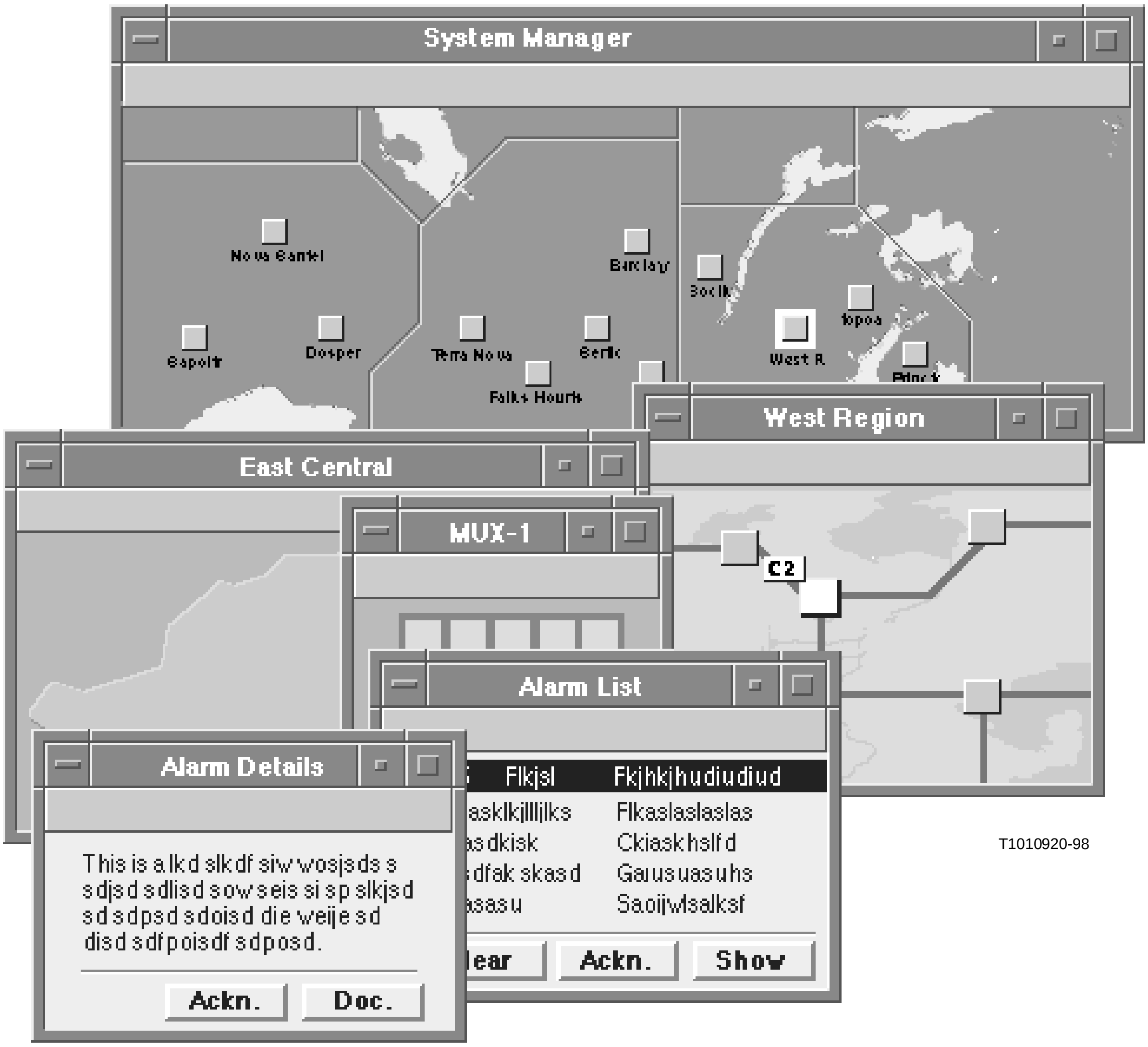
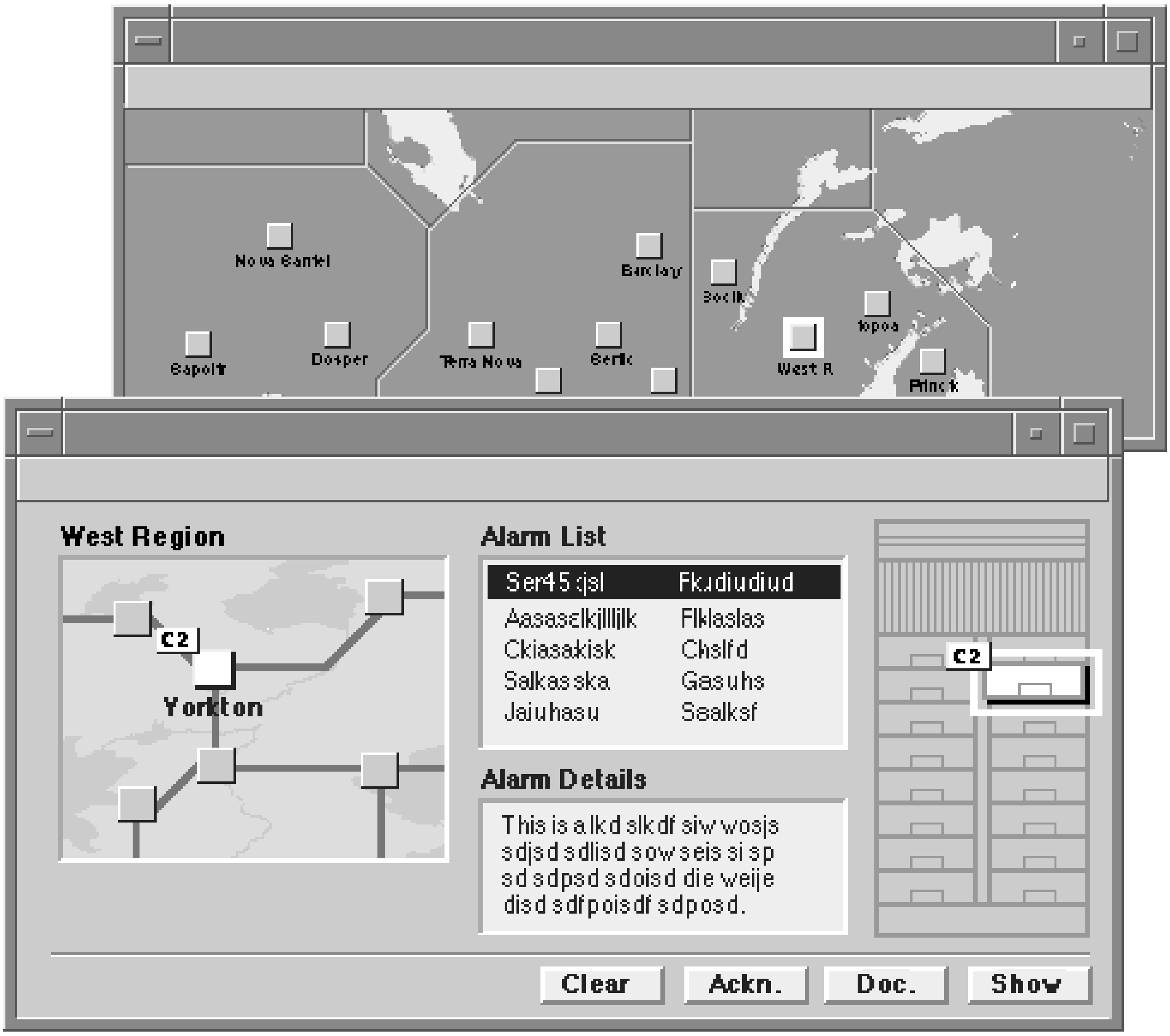


Figure 6.4-1/Z.361 – The software hierarchy exposed in the HCI

The example in Figure 6.4-2 shows the situation where the designer has taken care to assemble all the relevant information in one window.



T1010930-98

Figure 6.4-2/Z.361 – Flattened hierarchy at the HCI

6.5 Providing for multiple concurrent views

6.5.1 Background

Users engaged in network management tasks such as surveillance can benefit from being able to view different aspects of the network at the same time. These different views are valuable in problem-solving and depend not only on the task but on the users' understanding and experience.

6.5.2 Design guideline

Allow users to open and manipulate concurrent views of the network.

6.5.3 Rationale

When troubleshooting a fault, for example, a user may need to view details of several related network elements, and at the same time, maintain an overall view of the network.

Multiple views minimize the load on the user's short-term memory, thus reducing commission of memory-related errors.

Administrations should be able to change the ways in which the applications determine views, for example, what constitutes a region or a user's responsibility. Users should be able to change the layout of the views and what is presented in the views, for example, whether all network elements are displayed or just the transport elements and not the switching elements.

6.5.4 Examples of design implications

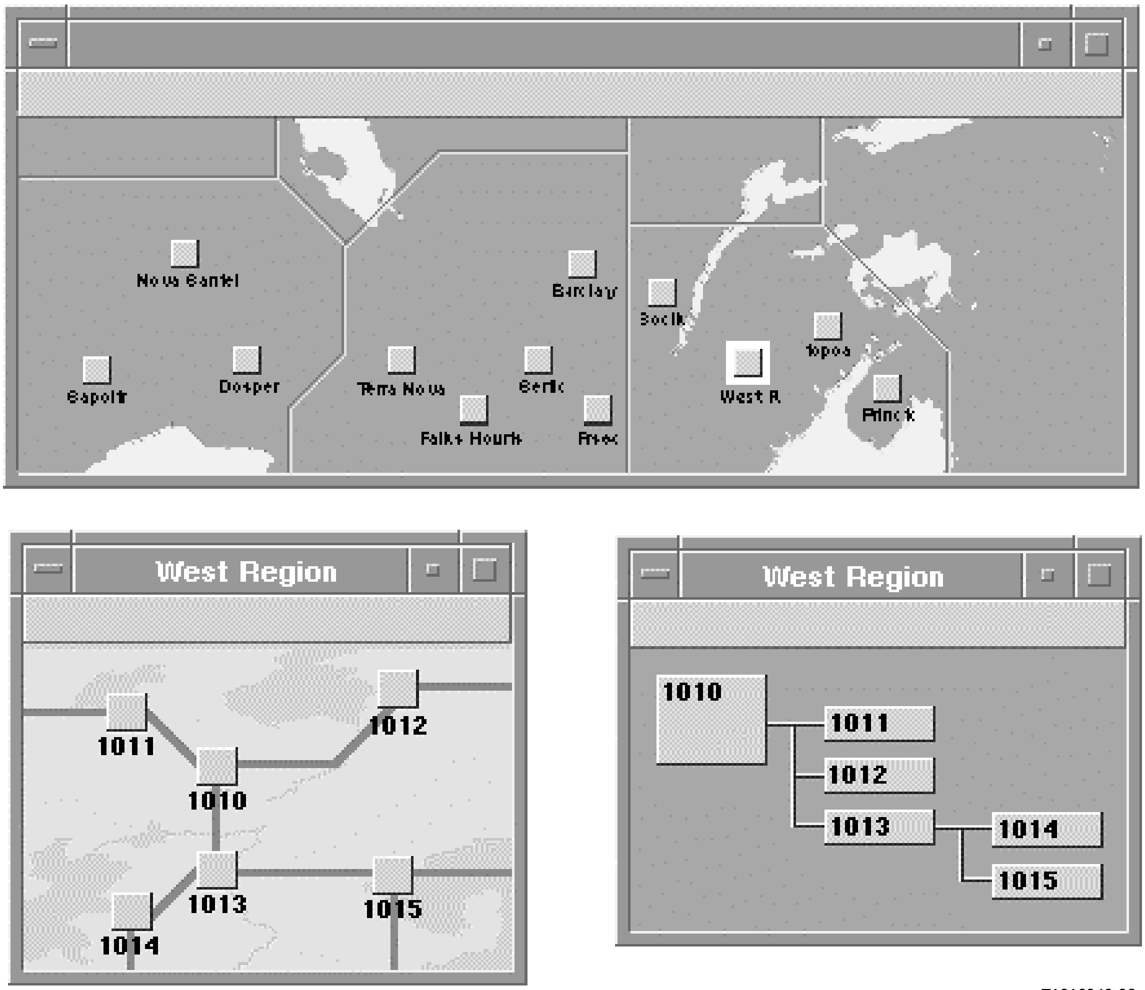
a) A user should be able to have open more than one view of the network at the same time.

b) Default parameters for windows' locations and sizes should be designed such that newly opened windows should not completely cover the previous active window.

c) Windows that are displaying network information should continue to have current up-to-date data displayed even when they are not the current user input focus.

6.5.5 Graphic example

The example in Figure 6.5-1 shows the availability of three different views, one being a more detailed view of a particular region. Different kinds of views such as logical views rather than physical views should also be made available to the user if required.



T1010940-98

Figure 6.5-1/Z.361 – Concurrent views of the network

6.6 Allowing for customization to support different user groups

6.6.1 Background

Administrations and users require the ability to adapt the interface to specific business requirements. At the same time, it should not be possible for users to so modify their displays that the efficiency or effectiveness of task performance is negatively affected.

6.6.2 Design guideline

Network management applications should allow several types of customization of their user interfaces to match administrative or user specific requirements.

6.6.3 Rationale

Regardless of how a given interface is constructed, there will be requirements to add or delete specific elements, choices, or options. These requirements result from differences in the operating environments in which the system is imbedded, differences in the vintage of equipment which is connected to the management platform, or differences in current operating practice and experience. Also, certain jurisdictions may allocate the tasks differently among their work forces. In addition, users may enhance their performance by customizing specific tools to accommodate task or individual requirements.

6.6.4 Examples of design implications

a) A system administrator or equivalent should be able to partition functionality and provide access controls for various user groups according to specific organization of the work.

b) Users should be able to customize their individual displays to reflect their level of expertise and preference of access to frequently performed tasks.

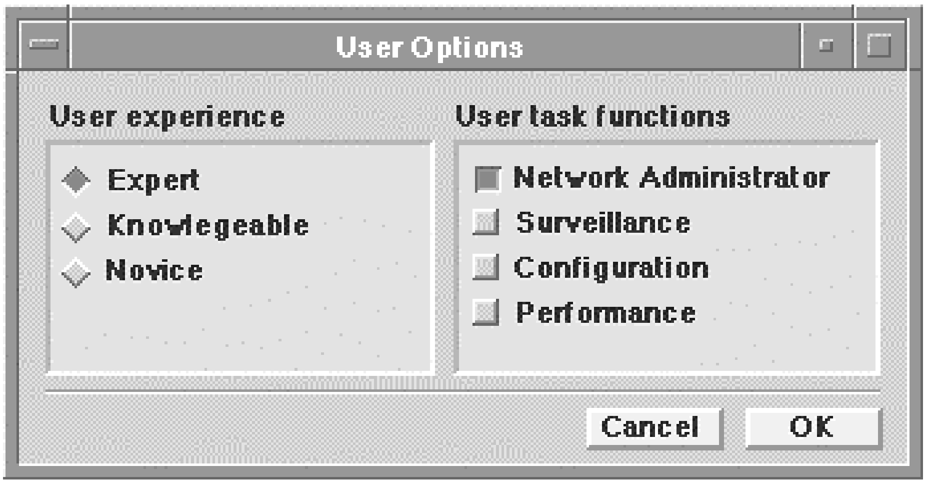
c) Users should not be able to alter the basic layout or presentation of their displays in ways that make them difficult or misleading to other users within the organization that perform the same or similar tasks. For example, users should not be able to alter colours used to signify alarmed elements.

d) The following aspects are examples of user interface customization requirements:

- Accommodation of native languages;
- Support for natural writing symbols and punctuation;
- Support for native conventions for date, currency, weights, numbers and addresses;
- Provision for specific work habits and the environment;
- Communication to users in natural and inoffensive ways;
- Sensitivity to the customer's culture;
- Specific colour requirements;
- Ability to filter out non-essential or unwanted information.

6.6.5 Graphic example

The example in Figure 6.6-1 shows two kinds of customization which are often required in telecommunications applications. They are: level of experience and task type.



T1010950-98

Figure 6.6-1/Z.361 – Customization of the HCI

ANNEX A

General HCI design objectives an principles

The following general HCI design objectives can be found throughout human factors literature and practice.

Consistency: Consistency of human-computer interface presentations and operations between and within applications helps users transfer and learning.

Empowering the user: Users should be given control over their applications, rather than having the application control the user.

Feedback: Users should receive feedback so they know what is happening in an application.

Efficiency: Applications should be designed to enable users of all skill levels to perform their work with a minimum number of steps.

Responsiveness: The interface should respond immediately or provide the user with information regarding the status of the system.

These are applicable to telecommunications applications interfaces as well as to other application domains. Part 10 of ISO 9241 presents a number of high-level dialogue principles. A dialogue for the purposes of ISO 9241-10 is "the interaction between a user and a system to achieve a particular goal".

The seven dialogue principles identified in ISO 9241-10 are:

- *Suitability for the task* – A dialogue is suitable for a task when it supports the user in the effective and efficient completion of the task.
- *Self-descriptiveness* – A dialogue is self-descriptive when each dialogue step is immediately comprehensible through feedback from the system or is explained to the user on request.
- *Controllability* – A dialogue is controllable when the user is able to initiate and control the direction and pace of the interaction until the point at which the goal has been met.
- *Conformity with user expectations* – A dialogue conforms with user expectations when it is consistent and corresponds to the user's characteristics, such as task knowledge, education and experience, and to commonly accepted conventions.
- *Error tolerance* – A dialogue is error tolerant if, despite evident errors in input, the intended result may be achieved with either no or minimal corrective action by the user.
- *Suitability for individualization* – A dialogue is capable of individualization when the interface software can be modified to suit the task needs, individual preferences and skills of the user.
- *Suitability for learning* – A dialogue is suitable for learning when it supports and guides the user in learning to use the system.

Documents called style guides provide information on how to use the tool kits to which they apply. Style guides provide much useful design information related to the recommended use and behavior of the basic interface elements or widgets such as buttons, menus and dialogues.

APPENDIX I

Bibliography

I.1 Related ISO standards

– ISO 9241-1:1997, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 1: General introduction.*

– ISO 9241-2:1992, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 2: Guidance on task requirements.*

– ISO 9241-3:1992, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 3: Visual display requirements.*

– ISO 9241-8:1997, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 8: Requirements for displayed colours.*

– ISO 9241-10:1996, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 10: Dialogue principles.*

– ISO 9241-14:1997, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 14: Menu dialogues.*

I.2 Other references

NIELSEN (J.): Usability Engineering, Academic Press, San Diego, 1994.

RUBIN (J.R.): Handbook of Usability Testing, John Wiley & Sons, New York, 1994.

APPENDIX II

Framework and HCI reference model from Recommendation Z.352

II.1 Design guidelines

This appendix relates the design guidelines in this Recommendation to the HMI reference model (see Figure II.1) as described in Recommendation Z.352.

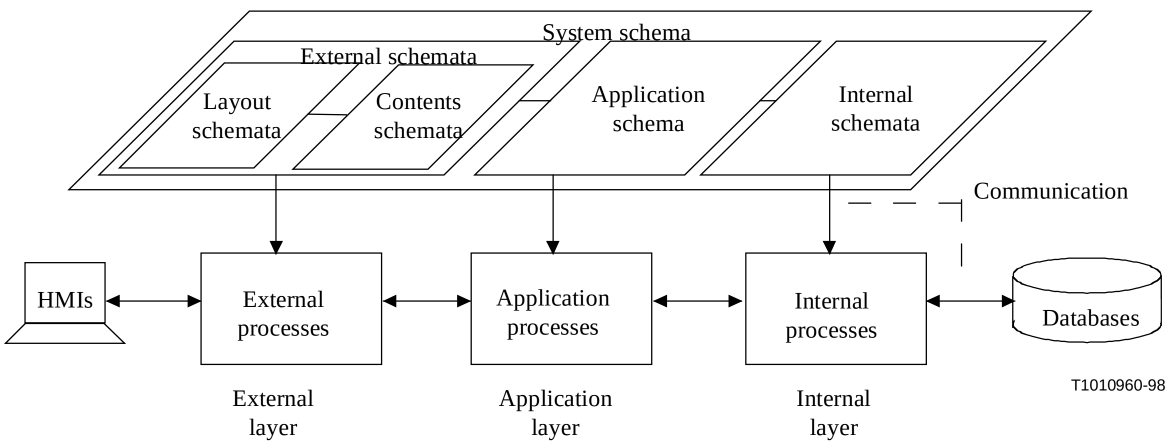


Figure II.1/Z.361

An overview of the guidelines and their relationships to the HMI reference model is provided in Table II.1.

Table II.1/Z.361 – Relationships between the guidelines and the HMI reference model

		Layout	Contents	Application
6.1	Avoiding the presentation of unnecessary information	L	C	A
6.2	Maintaining appropriate context	L	C	A, S
6.3	Providing redundant coding in all important graphic displays	L	C	A
6.4	Flattening the hierarchy		C	
6.5	Providing for multiple concurrent views	L	C	S
6.6	Allowing for customization to support different user groups	L	C	A

Guideline 6.1 relates to Layout schemata (L) through categorizing the contents of individual windows into three classes and prescribing different brightness of these classes. Co-ordination of colours and shapes can be provided in the Application schema (A).

In addition to prescription of the contents of windows, guideline 6.2 prescribes placement of information. This relates to the Layout schemata. However, the generic rules for placement of windows are stated in the common System schema (S) of the application.

Change of colour and other means of redundant coding in guideline 6.3 can be stated in the Layout schemata. However, text and graphic forms must typically be stated in the Application schema.

Guideline 6.4 relates to the design of the Contents schemata.

Guideline 6.5 relates to both the Layout schemata and System schema, as in guideline 6.2.

Guideline 6.6 relates to the means for customization of all schemata.

Note that both common and redundant presentation forms of information are defined in the Application schema. Only layout that can vary between different presentations, such as placement and sizes, are defined in the Layout schemata.

In a normal development process, the schemata are specified in the following sequence: Application, Contents and Layout schemata. When designing the first schemata, the designer has to observe the permissible contents of the succeeding schemata.

II.2 Terms and definitions

The following definitions are extracted from Recommendation Z.351. In addition, an example is provided for each definition, and new definitions are provided for System schema and System.

II.2.1 application schema: The application schema contains the data definitions, including constraints and derivation rules for corresponding population data of the application area. An application schema prescribes the terminology and grammar for an application area. Example: *Exchange* object class contains *Name* attribute with permissible values, *Number of lines* attribute etc.

II.2.2 external schema: The external schema contains the data definitions for the contents and layout of the corresponding population data at an actual human-machine interface. The data definitions can include constraints and derivation rules from the application schema. One external schema can contain data from one application schema only. Example: Mapping from EXCHANGE in the layout schema to Exchange in the corresponding contents schema, etc.

II.2.3 contents schema: The contents schema specifies the structure of the selected data and their relationships for a specific presentation. Each contents schema is contained in one external schema. The contents schema can contain specifications of permissible manipulations of the data in this presentation. Example: *Exchange* contains *name*.

II.2.4 layout schema: The layout schema specifies the way in which the data are to be presented to the human user. Each layout schema is contained in an external schema. Example: EXCHANGE in Arial font in line 3 column 7, Name in Arial font in line 4 column 7, and repeating value fields of length 8, adjusted to the right, in Times New Roman in line 9, etc.

II.2.5 human-machine interface: The human-machine interface is defined to consist of the presentation and manipulation of the HMI population data, the HMI external schemata and the HMI application schema for an application. Example:

EXCHANGE

Name

Ottawa

II.2.6 process: A process implements the enforcement of the rules found in a schema on the data instances in a corresponding population. Example: Value control of Ottawa.

II.2.7 schema: A schema contains the data definitions, including constraints and derivation rules for the corresponding population data. Example: See application, contents and layout schemata.

II.2.8 population: A population contains the data instances according to the rules expressed in a corresponding schema. Example: Ottawa, Montreal, Toronto.

II.2.9 system schema: A system schema contains specifications that are common for several schemata or mappings between schemata, such as directory information of the system, access rights, distribution and configuration specifications, style guides and common principles.

II.2.10 system: An integrated HMI system is an association of HMI data, functions and operations which ensures consistency of the information presented to the end user at any moment of time.

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure
Series Z	**Languages and general software aspects for telecommunication systems**