



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Z.352

(03/93)

MAN-MACHINE LANGUAGE

DATA ORIENTED HUMAN-MACHINE INTERFACE SPECIFICATION TECHNIQUE – SCOPE, APPROACH AND REFERENCE MODEL

ITU-T Recommendation Z.352

(Previously “CCITT Recommendation”)

FOREWORD

The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the International Telecommunication Union. 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, established the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

ITU-T Recommendation Z.352 was prepared by the ITU-T Study Group I (1988-1993) and was approved by the WTSC (Helsinki, March 1-12, 1993).

NOTES

1 As a consequence of a reform process within the International Telecommunication Union (ITU), the CCITT ceased to exist as of 28 February 1993. In its place, the ITU Telecommunication Standardization Sector (ITU-T) was created as of 1 March 1993. Similarly, in this reform process, the CCIR and the IFRB have been replaced by the Radiocommunication Sector.

In order not to delay publication of this Recommendation, no change has been made in the text to references containing the acronyms "CCITT, CCIR or IFRB" or their associated entities such as Plenary Assembly, Secretariat, etc. Future editions of this Recommendation will contain the proper terminology related to the new ITU structure.

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

© ITU 1994

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

CONTENTS

	Page
1 Scope of specification technique.....	1
2 Approach.....	1
2.1 Data versus functions.....	1
2.2 Consequences	1
2.3 Data design	3
3 Reference model.....	3
3.1 Overview	3
3.2 Scope	4
3.3 Subdivision of the external layer	6
Annex A – Guidelines for HMI developers	6
A.1 Introduction	6
A.2 Method.....	7
A.3 Data design	10
Appendix I – Additional guidelines for data designs	12
I.1 General.....	12
I.2 Design of classes.....	12
I.3 Design of identifiers	16
I.4 Derived data and level of detail.....	21

.....

SUMMARY

This Recommendation provides a framework for specifying harmonized HMIs.

The Recommendation points out the importance of identifying the right application area and the great impact this choice will have on the final design of the HMI. Therefore, a data oriented approach, rather than a functional approach, is proposed.

A three-layered reference model for HMI is introduced and explained. This reference model allows the centralization of data definitions for HMI and derivation of external presentations in a harmonized manner.

Annex A provides guidelines for how to carry out development of HMIs in accordance with the chosen reference model and formalism. The development method is not a waterfall-type development process. This annex also provides a framework for design of data seen at the HMI.

Appendix I provides additional guidelines for data design.

The mapping between the HMI reference model and the TMN functional architecture is under study.

Areas that this Recommendation has relations with and impact on, include:

- mapping to TMN specifications,
- software standardization.

Recommendation Z.352**DATA ORIENTED HUMAN-MACHINE INTERFACE
SPECIFICATION TECHNIQUE – SCOPE, APPROACH AND REFERENCE MODEL***(Helsinki, 1993)***1 Scope of specification technique**

The objective of this specification technique is to specify the data seen at the HMI and the grammar for these data. It is not the objective to define how the telecommunication networks operate internally.

The specification of the data is used by the end users prior to the implementation:

- to assess what an application is about;
- to ensure that its grammar is correctly understood;
- to ensure that its terminology is well chosen; and
- as a central part of a contract for the implementations.

The specifications should also be applicable as a means for:

- user guidance to the permissible and/or mandatory structure of the data and the explanations of the data;
- accessing the “instances” of the data for that application.

From this it follows that the specifications should be readable by and applicable to both designers and end users.

The designers can be both HMI designers and software designers. The end users can be both OA&M operators inside the Administrations and external customers administering their own services.

2 Approach**2.1 Data versus functions**

Functional specifications identify and decompose the tasks being performed in different functional areas. This decomposition of activities does not separate the objects that the functions apply to and how these objects are manipulated.

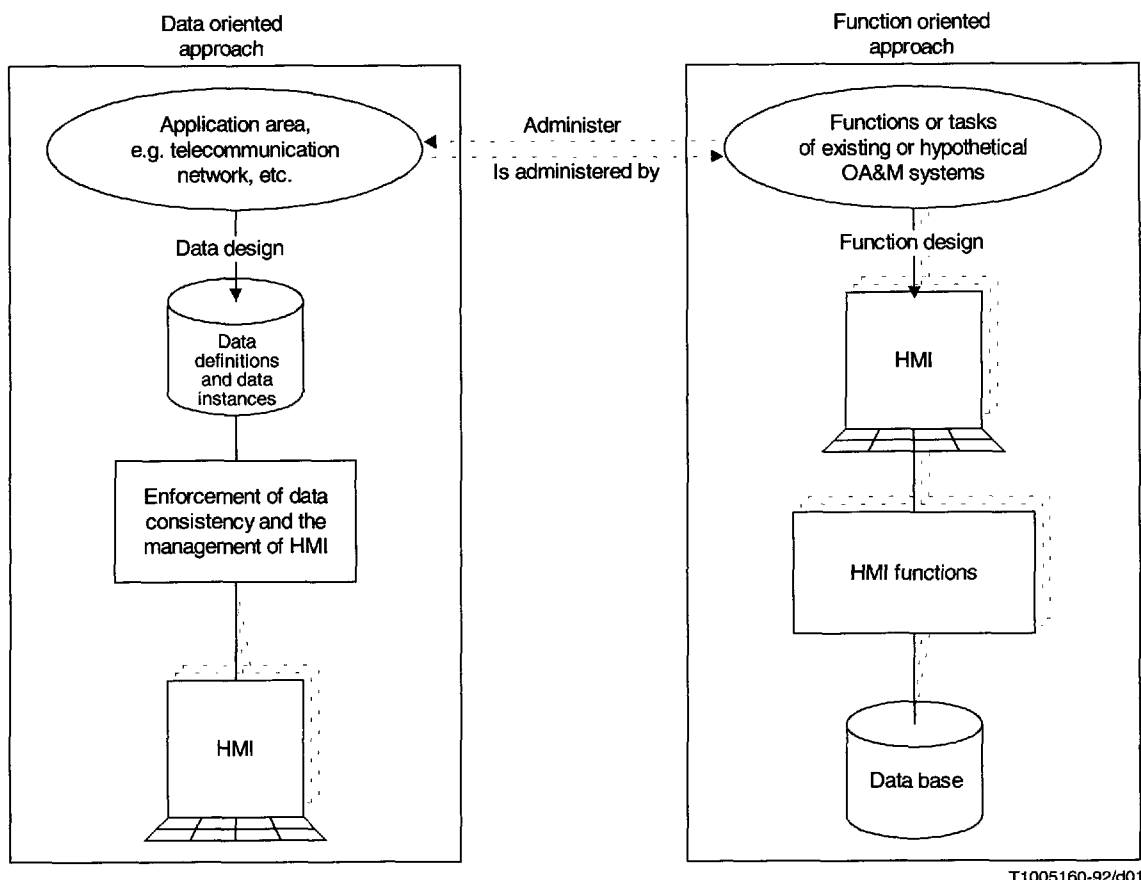
Data oriented approaches focus on identifying the data to which the functions apply, independently of the functions themselves. The means of manipulation on the data can most often be defined in a generic way – independently of the different types of data which are identified. It is very important to recognise that the two approaches tend to study two different universes of discourses (see Figure 1). Function oriented approaches study and decompose the tasks going to be performed in an organization. Data oriented approaches study and decompose the application area being administered by the organization. As a result of the data oriented approach a more coherent view of data is likely to be available for presentation to the human user.

2.2 Consequences

The implications of choosing one out of the two approaches are profound. The functions or tasks of any organization can change more rapidly than the data manipulated by these functions or tasks.

The function oriented approaches tend to analyse an old or hypothetical information system. This unavoidably conserves both existing characteristics and the designers' preconceptions into the design of the future system.

This observation has several further implications.



NOTES

- 1 In the function oriented approach the analysts study and decompose the tasks performed by the organization. For each task the human-machine interfaces are identified and specified. This is illustrated by the arrow labelled "function design" in the right hand part of the figure. The final function oriented system is implemented by developing one process for each individual HMI function, using one or several common data bases.
- 2 In the data oriented approach the analysts study and design common data definitions for the entire information system. This is illustrated by the arrow labelled "data design" in the left hand part of the figure. The processes that enforce consistency of the data can be made common for all data. The HMI does not have to be separated for each task, but can be common for several tasks. Note that the two approaches tend to study two different universes of discourse, as depicted by the two ovals in the upper part of the figure.

FIGURE 1/Z.352
Alternative approaches to HMI development

In the function oriented approaches the sequences of tasks to be performed are frozen into the implemented system; while the data oriented approaches provide more flexibility for changes in the sequences of tasks.

Both training and usage of systems designed according to the two different approaches tend to be very different. In the function oriented designs the users learn and use the different functions of the system and learn only implicitly about the universe of discourse, i.e. application area, and the structure of the data being manipulated by the functions. In the data oriented approaches the emphasis is put on understanding the application area being administered by the system and defined by the structure of the data of the system; the training on tasks to be performed is given less priority.

While function analysis can help to identify needs for data, function analysis does not help to identify data for additional functions, for new users or to identify unrelated data. Therefore, function analysis can constrain rather than support innovation and the questioning of existing data definitions. While the decomposition of functions can be an aid to the identification of needs for data, this decomposition is not of relevance in the final documentation of the data. Nevertheless, it may be useful to record the functional needs and motivation which led to the identification of data along with the definition of those data.

In the function oriented approaches the data are identified at the lowest level of decomposition of the functions. The identified data can be inputs to or outputs from the different functions. The identification of data separately in each function does not address and question the overall structure, design and definition of the data. Hence, the data definitions can continue to exist unharmonized and defined in such a way that they are unknowingly inefficient in use to the end users.

In the function oriented approaches separate design phases address the grouping of data into logical file structures, appropriate for the functions to be performed. These design phases take into account the access to and manipulation of data, but are usually not capable of reconsidering the overall design of data for that application.

2.3 Data design

In the data oriented approaches the data are identified independently of functions. The analysts and designers:

- observe the application area being administered by the organization;
- design data definitions for that application area.

The primary objective of the data oriented approaches is to design efficient data definitions for the chosen application area. For the design of data, a deep knowledge of the application area being administered is required and must be acquired.

The data definitions are important, because they can help the organization understand and perceive its functions and how it can perform its tasks.

3 Reference model

3.1 Overview

The data oriented approach allows for partitioning all HMI specifications and all software into a layered architecture:

- The external layer handles the presentation and manipulation of data. It also handles mapping to and subsetting of the application data.
- The application layer is that layer of the HMI reference model which is concerned with the definition of data and their behaviour.
- The internal layer is outside the scope of HMI. The layer is supposed to take care of the internal storage, accessing, implementation and communication of data and their behaviour.

Each layer of the layered architecture is partitioned into schemata, processes and populations. Data of each layer are mapped to data of adjacent layers only. A schema contains the data definitions, including constraints and derivation rules for the corresponding population data. A population contains the data instances which are enforced by a process according to the rules expressed in a corresponding schema. Only some aspects of the populations [HMI(s) and Data base(s)] are depicted in Figure 2. A process implements the enforcement of the rules found in a schema on the data instances in a corresponding population.

The application layer is a centralized resource of application data and their behaviour. The specification of the common terminology and grammar of all HMI data is not dispersed into several external functions, but is non-redundantly defined in the application schema.



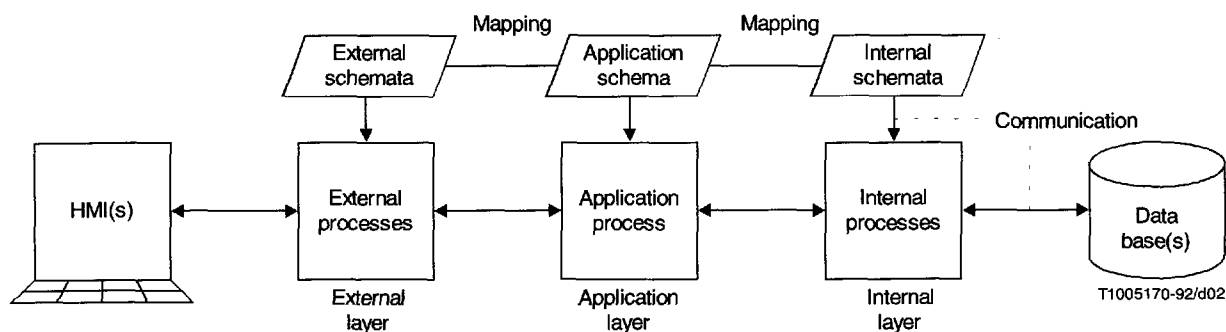


FIGURE 2/Z.352

The three-layered architecture**3.2 Scope**

The scope of the reference model of the HMI work is illustrated in Figure 3. The three-layered architecture is used only to define and illustrate the scope of the HMI and is not intended to impose restriction on the implementation.

The three layers are:

- the HMI external layer;
- the HMI application layer;
- the HMI internal layer.

The HMI application layer is controlled by one application schema, being a centralized resource, containing the specification of the structure and the dynamic behaviour of all HMI data.

The HMI external layer can contain several external schemata. The external schemata specify the detailed concrete syntaxes (layout) at the HMI and the access rights for each presentation.

The mappings between the application schema and the external schemata state the chosen subsetting and derivation for each presentation.

The internal layer is outside the scope of HMI. The layer is supposed to take care of the internal storage, accessing, implementation and communication of data and their behaviour.

The end users primarily use (the presentation of) HMI population data (instances), according to the external schema specifications.

In order to know which presentations are available, the end users need access to the external schemata and their contents.

However, the end users also need access to the definition of the data, their behaviour and interrelations. This information is provided from the application schema.

The end users do not normally need access to the internal aspects of the systems. The scope of the HMI work is therefore defined as consisting of the presentation and manipulation of:

- the HMI population data;
- the HMI external schemata;
- the HMI application schema.

The end users' requirements on the syntax and behaviour of the presentation of the HMI population data give requirements for the (presentation of the) external schemata. The (presentation of the) external schemata give requirements for the (presentation of the) application schema, which must be capable of specifying (the union of all) HMIs.