

---

---

**Information technology — Measurement  
and rating of performance of  
computer-based software systems**

*Technologies de l'information — Mesurage et gradation de la performance  
des systèmes de logiciels d'ordinateurs*

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

[ISO/IEC 14756:1999](https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999)

[https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-  
bfb4ba884527/iso-iec-14756-1999](https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999)



## Contents

Foreword .....	v
Introduction.....	vi
Section 1: General.....	1
1 Scope .....	1
2 Conformance.....	3
3 Normative reference.....	3
4 Definitions.....	4
5 Abbreviations and symbols .....	7
5.1 Abbreviations .....	7
5.2 Symbols .....	8
Section 2: Principles of measurement and rating.....	10
6 The measurement.....	10
6.1 Configuration requirements .....	10
6.2 User emulation.....	10
6.2.1 Random user behaviour.....	10
6.2.2 Remote terminal emulator.....	10
6.2.3 Workload parameter set .....	11
6.2.4 Parameter set for proving the accuracy of the user emulation .....	11
6.3 The measurement procedure .....	12
6.3.1 The time phases of the measurement procedure .....	12
6.3.2 Writing a measurement logfile .....	13
6.3.3 Writing a computation result file.....	13
6.4 Proof of validity of the measurement .....	13
6.4.1 Proof of the CBSS's computational correctness .....	13
6.4.2 Proof of the remote terminal emulator's accuracy .....	13
6.4.3 Proof of the measurement result's statistical significance .....	13
7 Calculation of the performance values of the SUT.....	14
7.1 Mean execution time.....	14
7.2 Throughput .....	14
7.3 Timely throughput .....	14
8 Basic data for rating .....	14
8.1 User requirements .....	14
8.2 The reference environment for rating software efficiency.....	14
8.2.1 Reference environment for assessing application software efficiency.....	15
8.2.2 Reference environment for assessing system software efficiency .....	15

© ISO/IEC 1999

All rights reserved. Unless otherwise specified, 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 publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland  
Printed in Switzerland

9 Rating the performance values .....	15
9.1 Computing the performance reference values .....	15
9.1.1 Mean execution time reference values .....	15
9.1.2 Throughput reference values .....	15
9.2 Computing the performance rating values .....	15
9.2.1 The mean execution time rating values .....	15
9.2.2 Throughput rating values .....	15
9.2.3 The timeliness rating values .....	16
9.3 Rating the overall performance of the SUT .....	16
9.4 Assessment of performance.....	17
9.4.1 The steps of assessment process .....	17
9.4.2 Weak reference environment .....	17
Section 3: Detailed procedure for measurement and rating .....	18
10 Input requirements .....	18
10.1 The SUT description.....	18
10.1.1 Specification of the hardware architecture and configuration .....	18
10.1.2 Specification of the system software configuration .....	18
10.1.3 The application programs .....	19
10.1.4 Additional software required for the measurement run.....	19
10.1.5 The stored data.....	19
10.1.6 Additional information for proof .....	19
10.2 The workload parameter set.....	19
10.2.1 The activity types .....	19
10.2.2 Activity input variation.....	20
10.2.3 The task types with timeliness function and task mode.....	20
10.2.4 The chain types and their frequencies.....	21
10.2.5 Preparation times mean values and their standard deviations .....	21
10.3 Input for measurement validation.....	22
10.3.1 Correct computation results.....	22
10.3.2 Variation of input data and its resulting output.....	22
10.3.3 Criteria for precision of working of the RTE .....	22
10.3.4 Criteria for statistical validity of results .....	22
11 The measurement.....	22
11.1 The measurement procedure .....	22
11.2 Individual rating interval .....	23
12 Output from measurement procedure.....	25
12.1 Measurement logfile .....	25
12.2 Computation result file .....	25
13 Validation of measurements .....	26
13.1 Validation of the computational correctness of the SUT .....	26
13.2 Validation of the accuracy of the RTE .....	26
13.2.1 Validity test by checking the relative chain frequencies .....	26
13.2.2 Validity test by checking the preparation times .....	26
13.3 Validation of the statistical significance of the measured mean execution time.....	27
14 Calculation of the performance values of the SUT .....	28
14.1 Mean execution time .....	28
14.2 Throughput .....	28
14.3 Timely throughput.....	28

- 15 Rating the measured performance values of the SUT ..... 29
  - 15.1 Specification of rating level ..... 29
  - 15.2 Computing performance reference values ..... 29
    - 15.2.1 Mean execution time reference values ..... 29
    - 15.2.2 Throughput reference values ..... 29
  - 15.3 Computing rating values ..... 29
    - 15.3.1 Computing mean execution time rating values ..... 29
    - 15.3.2 Computing throughput rating values ..... 30
    - 15.3.3 Computing timeliness rating values ..... 30
  - 15.4 Rating ..... 30
    - 15.4.1 Mean execution time rating ..... 30
    - 15.4.2 Throughput rating ..... 31
    - 15.4.3 Timeliness rating ..... 31
    - 15.4.4 Overall rating ..... 31
- Annex A (normative) Specification of the RTE's basic functions ..... 32
- Annex B (normative) Additional calculation formulas ..... 33
- Annex C (normative) Format of the workload description ..... 41
- Annex D (normative) Format of the logfile ..... 45
- Annex E (informative) Utility programs ..... 46
- Annex F (informative) Examples of workloads ..... 48

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

[ISO/IEC 14756:1999](https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999)  
<https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999>

## Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 14756 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software engineering*.

Annexes A to D form an integral part of this International Standard. Annexes E and F are for information only.

## iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 14756:1999](https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999)

<https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999>

## Introduction

In both the planning and using of data processing systems, the speed of execution is a significant property. This property is influenced greatly by the efficiency of the software used in the system. Measuring the speed of the system as well as the influence of the efficiency of the software is of elementary interest.

In order to measure the influence of software on the time behaviour of a data processing system it is necessary to measure the time behaviour of the whole system. Based on the metrics of the measurement procedure proposed in this standard it is possible to define and to compute the values of the time efficiency of the software.

It is important that time behaviour characteristics are estimated in a reproducible way. Therefore it is not possible to use human users in the experiment. One reason is that human users cannot reproduce longer phases of computer usage several times without deviations in characteristics of usage. Another reason is that it would be too expensive to carry out such experiments with human users if the job or task stream comes from many users. Therefore an emulator is used which emulates all users by use of a second data processing system.

This means that measurement and rating of performance according to this International Standard needs a tool. This tool is the emulator which shall work according to the specifications of this standard. It has to be proven that the emulator used actually fulfils these specifications.

All relevant details of this experiment are recorded in a logfile by the user emulator. From this logfile the values which describe the time behaviour (for instance response times and throughput values) can be computed. From these performance values the software efficiency rating values will be computed.

Not all of these values are always necessary to carry out a measurement and rating procedure. For instance if a simple workload having only a few interactive task types or only a simple sequence of batch jobs is used, then only a small subset of all terms and values which are defined is required. This method also allows the measuring and rating of a large and complex computer-based software system (CBSS) processing a complex job or task stream which is generated by a large set of many different users. As far as it is necessary the definitions include mathematical terms. This is in order to obtain an exact mathematical basis for the computations of performance and rating values and for checking the correctness of the measurement run and rating steps as well as for the (statistical) significance of the performance values and rating results.

The result of a measurement consists of the calculated performance values. These are throughput values and execution time values. The final result of performance assessment of a CBSS consists of the rating values. They are gained by comparing the calculated performance values with the user's requirements. In addition it is possible - if desired - to rate the performance values of the CBSS under test by comparing them with those of a reference CBSS (for instance having the same hardware configuration but another version of the application program with the same functionality).

The result of the rating procedure is a set of values, each being greater than, less than or equal to 1. The rating values have the meaning of "better than", "worse than" or "equal to" the defined requirements (or the properties of a second system under test used as a reference). The final set of rating values assesses each task type which are defined separately in the workload.

Annexes E and F contain software as well as special data that are not printable. Therefore they are delivered on the CD-ROM which constitutes this International Standard. A short overview is provided in both annexes.

# Information technology – Measurement and rating of performance of computer-based software systems

## Section 1: General

### 1 Scope

This International Standard defines how user oriented performance of computer-based software systems (CBSS) may be measured and rated. A CBSS is a data processing system as it is seen by its users, e.g. by users at various terminals, or as it is seen by operational users and business users at the data processing center.

A CBSS includes hardware and all its software (system software and application software) which is needed to realize the data processing functions required by the users or what may influence to the CBSS's time behaviour.

This International Standard is applicable for tests of all time constrained systems or system parts. Also a network may be part of a system or may be the main subject of a test. The method defined in this International Standard is not limited to special cases like classic batch or terminal-host systems, e.g. also included are client server systems or, with a broader comprehension of the definition of task, real time systems. But the practicability of tests may be limited by the expenditure required to test large environments.

This International Standard specifies the key figures of user oriented performance terms and specifies a method of measuring and rating these performance values. The specified performance values are those which describe the execution speed of user orders (tasks), namely the triple of:

- execution time,
- throughput,
- timeliness.

The user orders, subsequently called tasks, may be of simple or complex internal structure. A task may be a job, transaction, process or a more complex structure, but with a defined start and end depending on the needs of the evaluator. When evaluating the performance it is possible to use this International Standard for measuring the time behaviour with reference to business transaction completion times in addition to other individual response times.

The rating is done with respect to users requirements or by comparing two or more measured systems (types or versions).

Intentionally no proposals for measuring internal values, such as:

- utilisation values,
- mean instruction rates,
- path lengths,
- cache hit rates,
- queuing times,
- service times,

are given, because the definition of internal values depends on the architecture of the hardware and the software of the system under test. Contrary to this the user oriented performance values which are defined in this International Standard are independent of architecture. The definition of internal performance values can be done independently from the definition of user oriented performance values. They may be used and can be measured in addition to the user oriented performance values. Also the definition of terms for the efficiency with which the user oriented values are produced can be done freely. In addition this International Standard gives guidance on how to establish at a data processing system a stable and reproducible state of operation. This reproducible state may be used to measure other performance values such as the above mentioned internal values.

This International Standard focuses on:

- application software;
- system software;
- turn-key systems (i.e. systems consisting of an application software, the system software and the hardware for which it was designed);
- general data processing systems.

This International Standard specifies the requirements for an emulation (by a technical system - the so-called remote terminal emulator (RTE) - of user interactions with a data processing system. It is the guideline for precisely measuring and rating the user oriented performance values. It provides the guideline for estimating these values with the required accuracy and repeatability of CBSSs with deterministic as well as random behaviour of users. It is also a guidance for implementing a RTE or proving whether it works according to this International Standard.

This International Standard provides the guideline to measure and rate the performance of CBSS with random user behaviour when the accuracy and repeatability is required. It specifies in detail how to prepare and carry out the measurement procedure. Along with a description of the analysis of the measured values, the formulas for computing the performance value and the rating value, are provided.

This International Standard also gives guidance on:

- how to design a user oriented benchmark test using a:
  - \* transaction oriented workload, [ISO/IEC 14756:1999](https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-bfb4ba884527/iso-iec-14756-1999)
  - \* batch oriented workload,
  - \* or transaction and batch mixed workload.

It specifies:

- how to describe such a workload,
- how to perform the measurement procedure,
- how to rate the measured results.

This International Standard is of interest to:

- evaluators,
- developers,
- buyers (including users of a data processing system),
- system integrators

of CBSSs.

NOTE 1 The field of application of this International Standard may be extended to include the following aspects. Workloads fulfilling the specifications of this standard and having a sufficiently general structure may be used as standard workloads. They may be used to measure and rate performance of data processing systems used in specific fields. E.g. a standard workload for word-processing may be used to compare the time efficiency of different software products or different versions of the same product running on the same hardware system. Such a standard workload may also be used if always applying the same application software version and the same hardware to compare the efficiency of the system software. When applying the same application software and workload to different systems, consisting of hardware and system software, as normally sold by system vendors, the efficiency of the data processing systems may be compared with respect to the application and workload used.



## 2 Conformance

Rating a software system without comparing to another can be done following the rules of this International Standard by rating against user requirements. In case of comparing performance values developed through the use of this International Standard, the comparisons depends upon equivalent functions in the compared systems. The values are most useful when comparing different releases or platform versions of the same software system, or comparing software systems which are known to have equivalent functions, or comparing hardware by using software with equivalent functions. The values are not useful for comparing software systems which do not have known equivalency.

To conform to this International Standard the requirements in

- subclauses 6.1 and 10.1 for descriptions of the configuration including the system under test,
- subclauses 6.2 and 10.2 and annexes A and C for user emulation,
- subclauses 6.3, 11, 12 and annex D for measurement procedures,
- clauses 7, 14 and annex B for calculating performance values,
- clauses 8, 9 and 15 for rating procedures

shall be fulfilled.

For results of a measurement in addition all requirements in this International Standard shall be fulfilled and the tests in 6.4 and - in more detail - in clause 13 shall be carried out successfully without any ensuing failures. It is the responsibility of the tester to submit proof of the results of the measurements done in accordance with this International Standard. Therefore the tester should supply additional documents of their own choice in addition to the documents requested in this International Standard, which are suitable to repeat the measurement by a third party to attain the same results.

ITeh STANDARD PREVIEW  
(standards.iteh.ai)

## 3 Normative reference

[ISO/IEC 14756:1999](#)

<https://standards.iteh.ai/catalog/standards/sist/4eec5597-8a94-4f58-91ac-134408181275/iso-iec-14756-1999>

The following normative document contains provisions which, through reference in this text, constitute provisions of this International Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent edition of the normative document indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 14598-1:1999, *Information technology — Software product evaluation — Part 1: General overview*.

## 4 Definitions

For the purposes of this International Standard, the following definitions apply.

**4.1 activity:** An order submitted to the system under test (SUT) by a user or an emulated user demanding the execution of a data processing operation according to a defined algorithm to produce specific output data from specific input data and (if requested) stored data.

**4.2 activity type:** A classification of activities defined by the execution of the same algorithm.

**4.3 chain:** One or more tasks submitted to the SUT in a defined sequence.

**4.4 chain type:** A classification of chains which is defined by the sequence of tasks types.

### NOTES

2 The emulated users submit only chains of specified chain types to the SUT.

3 The mathematical symbol for the current number of the chain type is  $l$ . The mathematical symbol for the total number of chain types is  $u$ .

**4.5 computer-based software system (CBSS):** A software system running on a computer.

NOTE 4 A CBSS may be a data processing system as seen by human users at their terminals or at equivalent machine-user-interfaces. It includes hardware and all software (system software and application software) which is necessary for realizing data processing functions required by its users.

**4.6 emulated user:** The imitation of a user, with regard to the tasks he submits and his time behaviour, realized by a technical system.

**4.7 execution time:** The time which elapses between task submission and completion.

### NOTES

5 In case of a task is representing a batch job the execution time is the elapsed time for the completion of the job. In case of an interactive task the execution time is the response time (submit until complete response) of the task.

6 The reader should notify that the definition (and therefore the measurement and rating) of the begin and end of the execution time does not depend on the task mode. In case of "NO WAIT" task mode for the following task it is possible and solved by the method described in this Intentional Standard, that the execution time of the following task may overlap one or more preparation and execution times of subsequent tasks of the same emulated user.

**4.8 mean execution time:** The mean value of all execution times of tasks of the  $j$ -th task type which were submitted within the rating interval.

NOTE 7 The mathematical symbol is  $T_{ME}(j)$  corresponding to the  $j$ -th task type.

**4.9 mean execution time rating value:** The quotient (corresponding to the  $j$ -th task type) of the mean execution time reference value and the measured mean execution time.

NOTE 8 The mathematical symbol is  $R_{ME}(j)$  corresponding to the  $j$ -th task type.

**4.10 mean execution time reference value:** The mean execution time maximally accepted by the emulated user. It shall be computed from the workload parameter set (see 9.1.1, 15.2.1 and clause B.1).

### NOTES

9 The mathematical symbol is  $T_{Ref}(j)$  corresponding to the  $j$ -th task type.

**4.11 observation period:** The time interval, where the measurement procedure is observed for collecting (logging) measurement results for rating or validation, consisting of the rating interval and the supplementary run.

**4.12 preparation time:** The time which elapses before the task submission. The event of starting the preparation time depends on the definition of the task mode of the following task. The preparation time value is the random chosen representation of an distributed variable with a defined mean and a standard deviation. They depend on the task type of the following task and the type of the emulated user generating the task.

NOTES

10 The preparation time starts with the preceding task completion of the same emulated user if task mode of the following task equals 1, it starts with the preceding task submission if task mode equals 0, (see definition of task mode), regardless whether the preceding task belongs to the same chain or to the preceding chain.

11 The mathematical symbol of the mean preparation time is  $h(i,j)$  with its standard deviation  $s(i,j)$  corresponding to the  $i$ -th user type the  $j$ -th task type.

**4.13 rating interval:** The time interval of the measurement procedure from the time the SUT reaches a stable state of operation to the time the measurement results are fulfilling the required statistical significance.

NOTE 12 The mathematical symbol of the duration is  $T_R$ .

**4.14 relative chain frequency:** The relative frequency of using the  $l$ -th chain type by an emulated user of the  $i$ -th type.

NOTE 13 The mathematical symbol is  $q(i,l)$  corresponding to the  $i$ -th user type and the  $l$ -th chain type.

**4.15 remote terminal emulator (RTE):** A data processing system realizing a set of emulated users.

**4.16 stabilization phase:** The time interval of the measurement procedure when the RTE starts submitting tasks until the SUT reaches a stable state of operation.

**4.17 supplementary run:** The time interval of the measurement procedure from the time the measurement results fulfil the required statistical significance to the time when all tasks, which were submitted during the rating interval, are completed.

**4.18 system under test (SUT):** The parts of the CBSS to be tested. All components which may influence the SUT's time behaviour shall be part of the SUT and if the influence depends on some workload, this workload shall be represented by the RTE too. Except if an influence to the time behaviour is impossible or not evident the emulation of the parts beside may be omitted.

NOTE 14 The SUT may consist of hardware, system software, data communication features or application software or a combination of them. Testing a part of a system, means that all parts of the system, which may influence the time behaviour of the part to be tested, are an integral part of the SUT, e.g. testing the time behaviour of one host application beside others on the same host, the workload of all applications have to be defined and emulated by the RTE with a representative workload parameter set.

**4.19 task:** The combination of:

- a specific activity;
- a demanded execution time, defined by a specific timeliness function;
- a specific task mode.

**4.20 task completion:** Timely event when for a specific task the total output string or, in case of a set of output strings, all parts are completely received by to the emulated user or another instance. If the task does not submit an output to the user (i.e.: during the measurement: to the RTE) a functionality, producing an „artificial output“, indicating the task completion, may be added to the task for usage during the measurement.

NOTES

15 The time of task completion defines the end time of the preceding preparation time and the begin time of the execution time of the following task.

16 The correctness of the received output is validated by checking the computational result of the task (see 6.4.3, and in more detail 10.3.1 and 13.1).

**4.21 task mode:** Indication of whether the user's preparation time begins immediately with the task submission of the preceding task (*value* = 0, i.e. "NO WAIT") or begins when the preceding task has been completed (task completion) (*value* = 1, i.e. "WAIT").

## NOTES

- 17 This is not standard usage mode of "Dialog" or "Batch" in UNIX based systems.
- 18 The mathematical symbol of the task mode is  $M(j)$  corresponding to the  $j$ -th task type.

**4.22 task submission:** Timely event when the input string is completely submitted from the emulated user to the SUT and the execution of the task may start, regardless if the SUT starts the execution immediately or not. The task submission is also indicated by the time when the action of the emulated user for this tasks ends. Following this International Standard, the task submission shall not be defined (and measured) by the event, when the SUT has received or interpreted the input string or when a receipt string, which may be send by the SUT after receiving and interpreting the input string, is received by the emulated user.

NOTE 19 Normally the task submission is defined internally by the submission of a special character (e.g. Carriage Return) or a character sequence at the end of the input string or at the end of several parts of the input string. Also it often happens that the task submission event is defined by the submission of the last character of an specified number characters in a string. For a classic batch task the task submission may be defined by the submission of the last character of the last string of the batch command sequence.

**4.23 task type:** A classification of tasks which is defined by the combination of:

- the activity type, or  
a set of activity types which are all belonging to an identical timeliness function and task mode;
- the timeliness function;
- the task mode.

The STANDARD PREVIEW  
(standards.iteh.ai)

## NOTES

- 20 Emulated users submit only these types of tasks to the SUT.
- 21 The mathematical symbol for the current number of the task type is  $j$ . The total number of task types is  $m$ .

**4.24 throughput:** The rate (i.e. the average number per time unit with respect to the rating interval) of all tasks of a task type submitted to the SUT.

## NOTES

- 22 The mathematical symbol is  $B(j)$  corresponding to the  $j$ -th task type.
- 23 Usually throughput is defined by the rate of terminated tasks during a period of time. In order to specify a supplementary run and not a "heads run" in this standard, throughput is defined in this standard by the rate of tasks submitted. Nevertheless the correct task completion is validated by checking the computational result of the task.

**4.25 throughput rating value:** The quotient (corresponding to the  $j$ -th task type) of the (actual) throughput and the throughput reference value.

NOTE 24 The mathematical symbol is  $R_{TH}(j)$  corresponding to the  $j$ -th task type.

**4.26 throughput reference value:** The minimum throughput required by the set of emulated users.

NOTE 25 The mathematical symbol is  $B_{Ref}(j)$  corresponding to the  $j$ -th task type.

**4.27 time class:** A time limit, combined with a relative frequency corresponding to the ratio of the number of tasks (of a specific task type) - with an execution time less than or equal to the corresponding time limit - to the total amount of tasks (of that particular task type), used for comparison with the execution time of a task (of that particular task type).

NOTE 26 The mathematical symbol for the total number of time classes of the  $j$ -th task type is  $z(j)$ ; the time limit is  $g_T(j,c)$  where  $j$  is the current number of the task type and  $c$  is the current number of the time class, running from 1 to  $z(j)$ ; the relative frequency is  $r_T(j,c)$  corresponding to the time limit  $g_T(j,c)$ .

**4.28 timeliness rating value:** The quotient (corresponding to the  $j$ -th task type) of the timely throughput and the total throughput.

NOTE 27 The mathematical symbol is  $R_{TI}(j)$  corresponding to the  $j$ -th task type.

**4.29 timeliness function:** A description of the user requirements with respect to the execution times of tasks of a specific task type. It consists of one or more time classes. The relative frequency of the time class with the highest time limit shall be 1.0 (i.e. = 100%).

#### NOTES

28 The timeliness function corresponding to the  $j$ -th task type consists of  $z(j)$  time classes, each time class consisting of the pairs of numbers  $g_T(j,c)$  and  $r_T(j,c)$ .

29 An example of a timeliness function with 2 classes ( $z(j) = 2$ ) is:

The completion of on-line transactions of a specific type have to be done within:

- 1,5 seconds by at least 90% of the transactions;
- 4,0 seconds by 100% of the transactions.

Up to 10% of the response times may have more than 1,5 seconds but not more than 4,0 seconds; response times greater than 4.0 seconds are not accepted. Assuming the timeliness function is defined for a task type with the number  $j = 4$  this would result in the following tabled timeliness function:

task type	time class	time class limit	relative class frequency
$j = 4$	$c = 1$	$g_T(4,1) = 1,5 \text{ sec}$	$r_T(4,1) = 0,9$
$j = 4$	$c = 2$	$g_T(4,2) = 4,0 \text{ sec}$	$r_T(4,2) = 1,0$

**4.30 timely throughput:** Throughput of all of those tasks whose execution times are accepted with respect to the timeliness function.

NOTE 30 The mathematical symbol is  $E(j)$  corresponding to the  $j$ -th task type.

**4.31 user:** A person (or instance) who uses the functions of a CBSS via a terminal (or an equivalent machine-user-interface) by submitting tasks and receiving the computed results.

**4.32 user type:** A classification of emulated users which is defined by the combination of:

- the relative frequencies of the use of chain types;
- the preparation times (mean values and their standard deviations).

## 5 Abbreviations and symbols

### 5.1 Abbreviations

For the purposes of this International Standard, the following abbreviations apply.

CBSS	computer-based software system
LAN	local area network
RTE	remote terminal emulator
SUT	system under test
WAN	wide area network

## 5.2 Symbols

For the purposes of this International Standard, the following symbols apply.

$ALPHA$	Confidence coefficient of mean execution time
$B(j)$	Throughput corresponding to the j-th task type
$B_{Ref}(j)$	Reference value of throughput corresponding to the j-th task type
$c$	Current number of a time class in a timeliness function
$d(j)$	Half width of the confidence interval of mean execution time corresponding to the j-th task type
$DELTA_h$	Required RTE precision indicator of mean preparation time. It is the maximally accepted relative difference of the actual mean preparation time to $h(i,j)$
$DELTA_q$	Required RTE precision indicator of chain frequency. It is the maximally accepted relative difference of the actual chain frequency to $q(i,l)$
$DELTA_s$	Required RTE precision indicator of the standard deviations of the preparation times. It is the maximally accepted relative difference of the actual standard deviations of the preparation time to $s(i,j)$
$DIFF_h$	RTE precision indicator of mean preparation time. It is the relative difference of the actual mean preparation time to $h(i,j)$ that has occurred during the observation period
$DIFF_q$	RTE precision indicator of chain frequency. It is the relative difference of the actual chain frequency to $q(i,l)$ that has occurred during the observation period
$DIFF_s$	RTE precision indicator of the standard deviations of the preparation times. It is the relative difference of the actual standard deviation of the preparation time to $s(i,j)$ that have occurred during the observation period
$E(j)$	Timely throughput corresponding to the j-th task type
$f(l,k)$	Task type of the k-th task in a chain of the l-th chain type
$g_T(j,c)$	Time limit of the c-th time class of the timeliness function corresponding to the j-th task type
$h(i,j)$	Mean preparation time of the j-th task type and corresponding to emulated users of the i-th user type
$i$	Current number of the user type
$j$	Current number of the task type
$k$	Serial number of a task in a sequence of tasks ('the k-th task of a particular type in the observation period' or 'the k-th task in a chain')
$K(j)$	Total amount of tasks of the j-th task type which were submitted within the rating interval
$l$	Current number of the chain type
$L_{chain}(l)$	Total number of tasks in the chain of the l-th chain type (i.e. length of the chain type)
$m$	Total number of different task types
$M(j)$	Task mode corresponding to the j-th task type
$n$	Total number of different user types
$N_{tot}$	Total amount of emulated users to be emulated by the RTE

$N_{user}(i)$	Total amount of emulated users of the $i$ -th user type
$p$	Total number of different timeliness functions
$q(i,l)$	Relative chain frequency; the relative frequency of using the $l$ -th chain type by an emulated user of the $i$ -th user type.
$r_T(j,c)$	Maximum accepted relative frequency of tasks of the $j$ -th task type whose execution times are less than or equal to the time limit $g_T(j,c)$
$R_{ME}(j)$	Mean execution time rating value corresponding to the $j$ -th task type
$R_{TH}(j)$	Throughput rating value corresponding to the $j$ -th task type
$R_{TI}(j)$	Timeliness rating value corresponding to the $j$ -th task type
$s(i,j)$	Standard deviation of the preparation time of emulated users of the $i$ -th user type and of tasks corresponding to the $j$ -th task type
$t_0$	Start time of the RTE, begin time of the stabilization phase
$t_1$	Begin time of the rating interval
$t_{1ind}$	Begin time of the individual rating interval of a specific emulated user
$t_2$	End time of the rating interval, begin time of the supplementary run
$t_{2ind}$	End time of the individual rating interval, begin time of the individual supplementary run of a specific emulated user
$t_3$	End time of the supplementary run
$t_{3ind}$	End time of the individual supplementary run of a specific emulated user
$t_4$	End time of the latest ending individual supplementary run
$t_{ET}(j,k)$	Execution time of the $k$ -th task (of the $j$ -th task type) which was submitted within the rating interval
$T_{ME}(j)$	Mean execution time corresponding to the $j$ -th task type
$T_{MR}$	Mean duration of the individual rating intervals of all emulated users, i.e. the mean of all $T_{Rind}$ values
$T_R$	Duration of the rating interval from $t_1$ to $t_2$ .
$T_{Rind}$	Duration of the individual rating interval of a specific emulated user from $t_{1ind}$ to $t_{2ind}$ .
$T_{Ref}(j)$	Reference value of mean execution time corresponding to the $j$ -th task type
$u$	Total number of different chain types
$w$	Total number of different activity types
$z(j)$	Number of time classes of the timeliness function corresponding to the $j$ -th task type