International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.1562

(03/2007)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

Internet protocol aspects – Quality of service and network performance

---

Framework for higher-layer protocol performance parameters and their measurement

ITU-T Recommendation Y.1562

ITU-T

International Telecommunication Union

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	**Y.1500–Y.1599**
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Numbering, naming and addressing	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899

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

ITU-T Recommendation Y.1562

Framework for higher-layer protocol performance parameters and their measurement

Summary

ITU-T Recommendation Y.1562 defines the parameters and methodology used to assess the service provided by higher-layer protocols. Parameters include performance parameters and availability parameters, such as high-layer packet transfer throughput, service delay, availability and so on. The methodology and parameters described in this Recommendation are applicable to networks supporting both IPv4 and IPv6.

Source

ITU-T Recommendation Y.1562 was approved on 1 March 2007 by ITU-T Study Group 12 (2005-2008) under the ITU-T Recommendation A.8 procedure.

Keywords

Availability, DHCP, FTP, higher-layer protocol, HTTP, performance, QoS.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. 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 Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 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 expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. 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, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

CONTENTS

ITU-T Recommendation Y.1562

Framework for higher-layer protocol performance parameters and their measurement

1 Scope

This Recommendation defines the end-to-end higher-layer performance parameters such as throughput, service availability and so on, and an associated evaluation methodology. The methodology described in this Recommendation relates to end-to-end higher-layer protocols based on IPv4 and IPv6, and the performance parameters described in this Recommendation are used to assess the service status of higher-layer protocols. The methodology and parameters described in this Recommendation are applicable to networks supporting both IPv4 and IPv6.

The node in the end-to-end model described in this Recommendation may be a host, router, switch or network device. The only requirement is that it should support a higher-layer protocol and test function.

The methodology described in this Recommendation can be used to examine the status of higher-layer protocols which provide service to the customer, and provide performance parameters to assess the service quality.

The performance parameters are intended to be used in planning and offering international higher-layer protocol service. The intended users of this Recommendation include carrier service providers, equipment manufacturers and end users. This Recommendation may be used by service providers in the planning, development, and assessment of IP service that meets user performance needs; by equipment manufacturers as performance information that will affect equipment design; and by end users in evaluating higher-layer protocol service performance.

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; 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. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.1000] ITU-T Recommendation G.1000 (2001), *Communications Quality of Service: A framework and definitions*.

[ITU-T G.1010] ITU-T Recommendation G.1010 (2001), *End-user multimedia QoS categories*.

[ITU-T I.350] ITU-T Recommendation I.350 (1993), *General aspects of quality of service and network performance in digital networks, including ISDNs*.

[ITU-T Y.1540] ITU-T Recommendation Y.1540 (2002), *Internet protocol data communication service – IP packet transfer and availability performance parameters*.

[ITU-T Y.1541] ITU-T Recommendation Y.1541 (2006), *Network performance objectives for IP-based services*.

[ITU-T Y.1560] ITU-T Recommendation Y.1560 (2003), *Parameters for TCP connection performance in the presence of middleboxes*.

3 Definitions

This Recommendation defines the following term:

3.1 measurement point (MP): A MP is located at the interface between the transmission layer and a higher layer on a host that separates the customer network from an attached transmission system, and at which a reference event can be observed and measured.

4 Abbreviations

This Recommendation uses the following abbreviations:

CESR	Connection Establishment Success Ratio
DHCP	Dynamic Host Configuration Protocol
FTP	File Transfer Protocol
HLAD	Higher-Layer Authentication Delay
HLDTD	Higher-Layer Data Transfer Delay
HLPT	Higher-Layer Packet Throughput
HLSD	Higher-Layer Service Delay
HLSRD	Higher-Layer Service Response Delay
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
MN	Measurement Node
MP	Measurement Point
NMS	Network Management System
SA	Service Availability
SSR	Service Success Ratio
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

5 Conventions

None.

6 Reference model for higher-layer protocols

6.1 Layered model of higher-layer protocols

Figure 6-1 illustrates the layered nature of high-layer protocols. The performance provided to higher-layer service users depends on the performance of the transmission layer, IP layer and other layers.

– Lower layers include the physical layer and link layer supporting the IP layer.

– The IP layer provides connectionless, best-effort forwarding service for the transmission layer. There are two versions of the IP protocol: version 4 and version 6. The performance of IPv4 is discussed in [ITU-T Y.1541]. The performance of IP layer for IPv6 is for further study.

– The transmission layer supported by the IP layer provides services for higher protocol layer. There are two kinds of transmission layer protocols: transmission control protocol (TCP) and user datagram protocol (UDP). The performance of TCP is discussed in [ITU-T Y.1560]; the performance of UDP is for further study.

– The higher layer provides reliable, end-to-end services for customers, for example, FTP, HTTP, DHCP, and so on.

NOTE 1 – The performance of higher protocol layers discussed in this Recommendation is applicable to both IPv4 and IPv6.

NOTE 2 – Performance interactions among these layers are for further study.

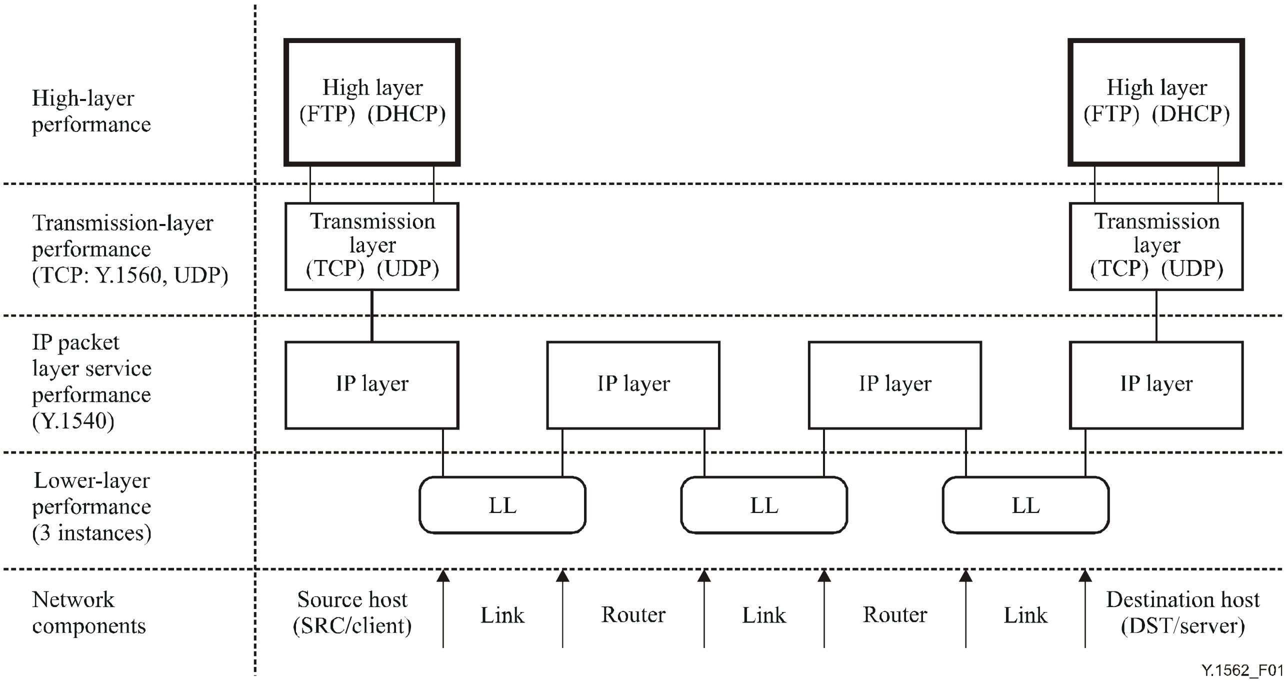


Figure 6-1 – Layered model for higher layer – Example

6.2 General NGN performance model

Some NGN applications will involve two or more clients. Familiar examples of application services involving two (or more) clients are instant messaging, remote control and monitoring, and real-time multi-party gaming. There will also be situations where the server/network interfaces must be considered in assessing application performance. This will be true in any situation where it is important to distinguish application server performance from network performance, e.g., when the application service provider and the network operator are in different organizations. In such situations, the performance models will involve four types of entities: the communication network, the application server, and two clients, distinguished by the roles they play in particular application service instances. The application server may sometimes be viewed as a user of the network service. It may also be important, in some situations, to specify and measure the contributions of individual network portions to the overall end-to-end performance of IP applications, or even to similarly partition an application server.

A very general NGN performance model that may reflect these considerations is illustrated in Figure 6-2. This model is consistent, at a very high level, with the NGN functional architecture model defined in ITU-T Rec. Y.2012. It will provide the flexibility to address many, but not all, IP application performance assessment situations. Since adding model interfaces complicates performance assessment, it will be important to use the simplest possible model in any particular application performance study.

In all applications of the NGN performance model, the reference events will be transfers of requests and responses across the depicted interfaces in accordance with standardized client/server protocols. As in the communication case, specific events will be defined in terms of interface protocol state transitions. A conceptual difference is that in the assessment of application performance, the relevant protocols will be higher "up the stack" (or stated differently, the performance-significant protocol information will be more deeply embedded in the packets observed at the physical interfaces). It will nevertheless be possible to extract the relevant information at such interfaces.

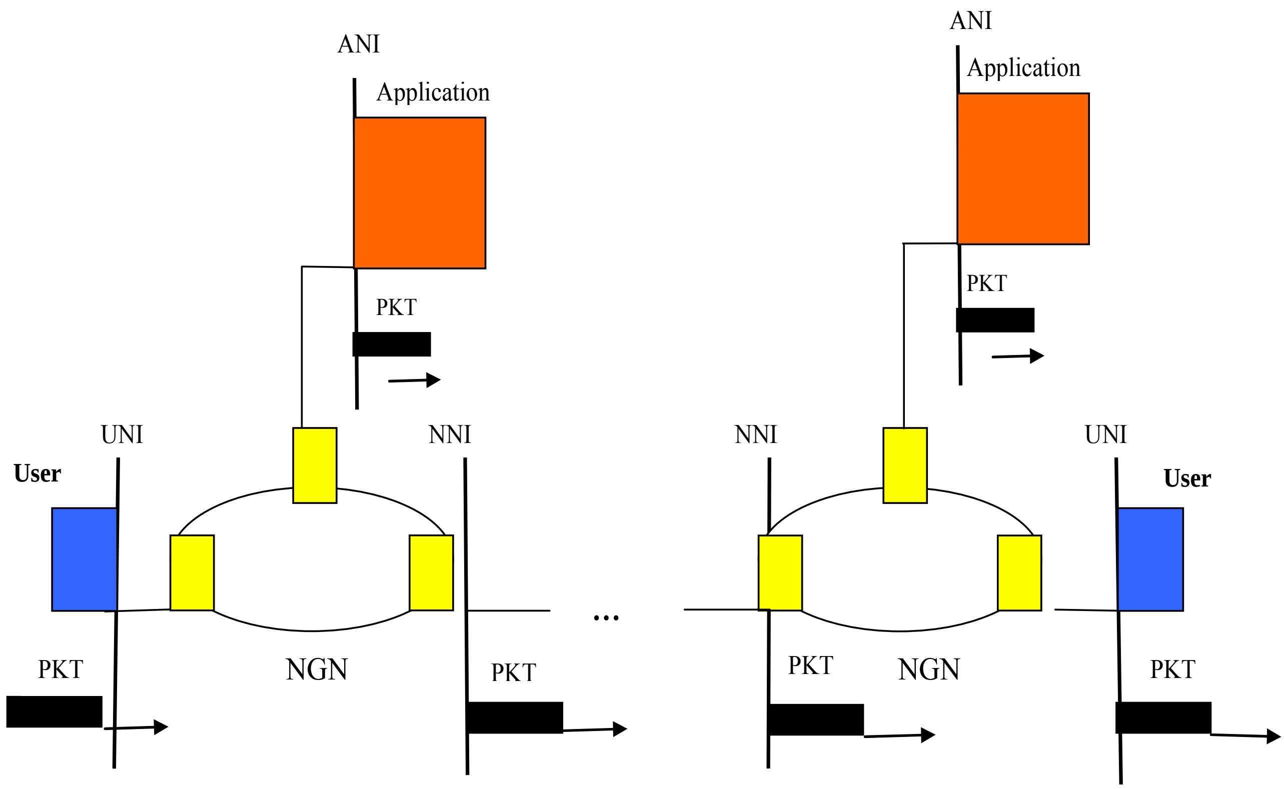


Figure 6-2 – General NGN performance model

As noted earlier, the functions performed by NGN IP applications will be more diverse, and typically much more complex, than those performed in traditional communication sessions. Application servers are typically designed to process input information in some defined way, to relate it to other input or stored information, or to provide additional, different, or restructured information in response to user requests. Servers frequently also store input information for later retrieval and use by the inputting client or another user. The set of higher-layer applications likely to be provided by NGNs is not bounded, and is therefore not subject to stable classification.

6.3 Reference model

Figure 6-3 illustrates the reference model for higher-layer protocols and the relationship between reference events and measurement points for a higher-layer protocol session.

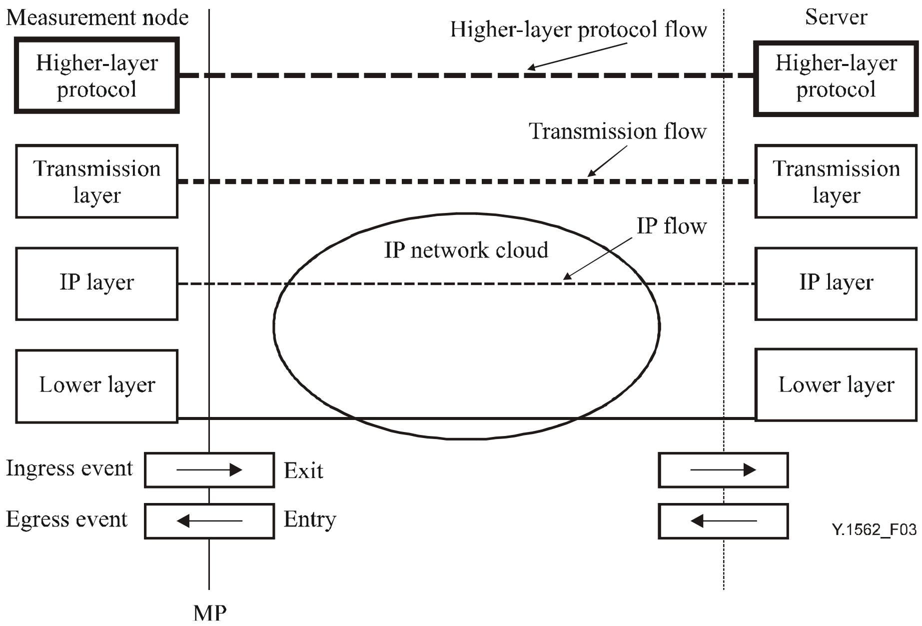


Figure 6-3 – Reference model

6.3.1 Network components

6.3.1.1 router: A device that can forward IP packets based on the content of their IP destination address field and routing information.

6.3.1.2 measurement node: The MN may be a computer, router, L3 switch or tester, which can initiate higher-layer protocol tests and record the result.

6.3.1.3 server: The server may be a computer, router, L3 switch or tester, which can provide higher-layer protocol service. In response to the request sent by MN, the server sends a reply to source/client.

6.3.2 Reference event

In the context of this Recommendation, the following definitions apply on a specified end-to-end higher-layer protocol service. The defined terms are illustrated in Figure 6-3.

A higher-layer protocol packet transfer event occurs when:

– a higher-layer protocol packet crosses a measurement point (MP); and

– the packet is valid according to the higher-layer protocol specification; and

– the source and destination address represent the IP address of the expected MN and server.

6.3.2.1 higher-layer protocol packet ingress event: A higher-layer protocol packet ingress event occurs when a higher-layer protocol packet crosses an MP entering IP network.

6.3.2.2 higher-layer protocol packet egress event: A higher-layer protocol packet egress event occurs when a higher-layer protocol packet crosses an MP exiting IP network.

6.3.3 Derivation of performance parameters

The next step of the process is to select and define a set of primary parameters that collectively characterize the network's speed, accuracy, and dependability in accomplishing each communication function. The primary parameters are defined on the basis of reference events that can be observed at the standardized interfaces. Formally, they are random variables on the sample space that distinguishes the possible function performance outcomes.

It is true that for many applications, there is a single "correct" response to each user input in each defined application state. For all applications with this property, it should be possible to define a set of application functions, similar conceptually to (although more complex and more numerous than) their communication counterparts (e.g., access, user information transfer, and disengagement), and to characterize the overall performance of the application in terms of the performance of these functions. The IP applications for which such systematic analysis is feasible will likely include file transfer, e-mail, instant messaging, presence/location, audio and video streaming, IPTV, remote control and monitoring, authentication and authorization, and domain name service, among others. The functions that comprise and support such applications can be described as single-valued functions.

A "trial performance" of any single-valued function can experience any one of the three categories of outcomes defined earlier (successful performance, incorrect performance, and non-performance), and the corresponding performance criteria (speed, accuracy, and dependability) will generally be of interest in assessing such application functions. An approach similar to that illustrated by the 3×3 matrix of [ITU-T I.350] could thus be used to identify and evaluate the associated performance parameters. Specifically, one could envision, for any application comprised of single-valued functions, a matrix listing the functions in rows and identifying, for each function, one or more performance parameters addressing each generic performance criterion. The importance of the various criteria (and any associated parameters) will of course depend on the function in question. Particular outcomes may not need parametric description in some applications, while in others, more detailed outcomes and associated parameters may need to be defined. The 3×3 matrix approach will nevertheless be useful as a starting point. It is simple, intuitive, and systematic, it encourages consistency, and it reduces the likelihood that any significant aspect of performance will be overlooked.

NGN will also support many important applications whose functions are not single-valued, or whose outcomes are non-discrete (i.e., continuous, unbounded, or qualitative in some respect). One example of such an application is web searching. Quantifying the performance of such applications will require a means of relating the server's "relative success" in performing the application functions with the client's satisfaction with the application service. Such measures may need to be subjective.

6.3.4 Application profiles

A key enabler of these solutions will be the standardization and use of representative event sequences, or application profiles, in active performance measurement. Such profiles will support NGN application performance assessment in the same way that "benchmark programs" have been used to assess the performance of computer systems. This approach will eliminate the variability caused by user dependence and ensure the validity of service provider performance assessments by acting as user surrogates with consistent (and internally measurable) performance characteristics. They will provide a generally accepted, operationally repeatable basis for overall application characterization, enabling time-based or transaction-based assessment of availability. They will extend the use and value of objective, perception-based performance metrics by controlling or measuring variables not otherwise addressed. They will facilitate the use of a systematic performance description process such as that described here.

6.3.5 Measurement session

Higher-layer protocols provide services for the customer, such as file transfer, address allocation, web-browsing and so on. Measurement of higher-layer protocol performance is used to characterize both IPv4 and IPv6 networks, and the measurement session is a procedure that the higher-layer protocol completes in a single service. Usually, the procedure includes a series of interactions between MN and server. The measurement of higher-layer protocol is an end-to-end session, where the end-points are MN and server.

In the measurement session, MN acts as a measurement point, which is responsible for initiating the measurement session, and sends a request for measurement to the server. After receiving the request message, the server responds to the request and sends a reply message to MN. When MN receives the reply message, MN will interact with the server to acquire the performance and availability parameters that are used to assess the networks. Figure 6-4 illustrates the measurement session.

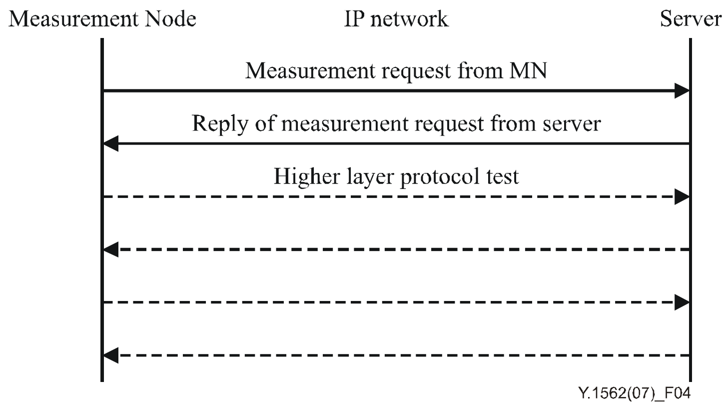


Figure 6-4 – Higher-layer protocol measurement session

7 Higher-layer protocol performance parameters

This clause describes the performance parameters of higher-layer protocols. Values of these performance parameters, which are obtained from MN, can be used to assess the status of higher-layer protocols.

7.1 Higher-layer service response delay (HLSRD)

Higher-layer service response delay is the time, $(t_2 – t_1)$, between when the MN sends the first packet which is the stimulus for a request packet ingress event (t_1) to the server, and receives a response from the server which stimulates a response packet egress event (t_2). MN sends the first packet to server at time t_1 and receives a response of service from the server at time t_2, where $(t_2 > t_1)$ and $(t_2 – t_1) \leq T_{max}$.

NOTE 1 – For connection-based service, such as TCP based, the higher-layer service response delay includes two parts: one is the connection establishment delay, and the other is the higher-layer protocol's response delay.

NOTE 2 – The higher-layer service response delay might include DNS resolve delay, which can be measured independently.

NOTE 3 – T_{max} is the maximum waiting time for a service response. The value of T_{max} must be set to distinguish between responses with a long (but possibly occurring) delay and failure conditions. Setting the value of T_{max} requires knowledge of network delays and application processing times.

Figure 7-1 illustrates the higher-layer service response delay.

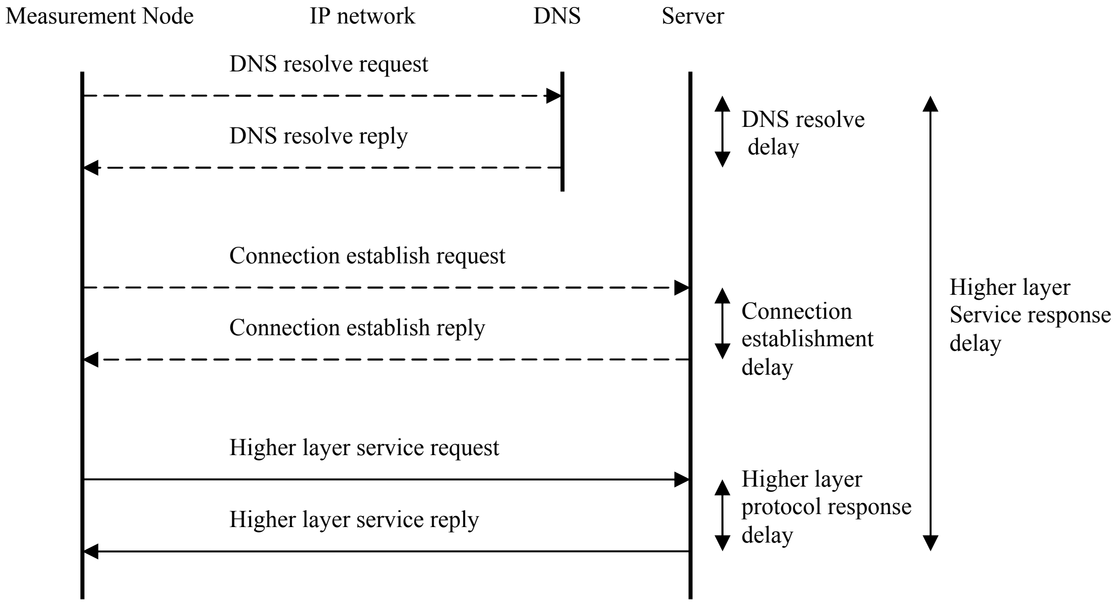


Figure 7-1 – Higher-layer service response delays

7.2 Higher-layer authentication delay (HLAD)

Some higher-layer services need to authenticate the user to verify whether the user has the right to use the service. The higher-layer authentication delay is the time, $(t_2 - t_1)$, between when the MN sends the user name and password to the server (ingress reference event) and receives the authentication response from the server (egress reference event). MN sends the user name and password to server at time t_1 and gets authentication response from server at time t_2, where $(t_2 > t_1)$ and $(t_2 - t_1) \leq T_{max}$.

7.3 Higher-layer data transfer delay (HLDTD)

The higher-layer data transfer delay is the time, $(t_2 - t_1)$, between when the MN starts to retrieve data from server and data retrieval is completed. MN starts to retrieve data from the server at time t_1 and data retrieval is completed at time t_2, where $(t_2 > t_1)$ and $(t_2 - t_1) \leq T_{max}$.

NOTE 1 – The time t_1 corresponds to the reference event when MN sends the request to server to retrieve data from server.

NOTE 2 – The time t_2 corresponds to the reference event when MN has received the last data packet from server.

7.4 Higher-layer service delay (HLSD)

The higher-layer service delay is the time, $(t_2 - t_1)$ between the start and end of the service. Service starts at time t_1 and ends at time t_2, where $(t_2 > t_1)$ and $(t_2 - t_1) \leq T_{max}$, where t_2 and t_1 correspond to reference events.

NOTE – In order to assess the service, the test itself should not introduce any additional impact on performance.

8 Higher-layer protocol availability parameters

8.1 Connection establishment success ratio (CESR)

For connection-based service, the connection must be established within a specified value of T_{max} before it can provide service. After the connection has been established, the MN can use the service that the server provided. The connection establishment success ratio is important for connection-oriented service.

The connection establishment success ratio is the number of successful connection establishments divided by the total number of trials.

NOTE – For connectionless service, this parameter has no meaning.

8.2 Service success ratio (SSR)

Service success ratio is a measure of how well the higher-layer protocol can provide service completely. If the higher-layer protocol has been measured for n times, and the service is successful for m times, then the service success ratio is m divided by n.

8.3 Service availability (SA)

The service availability describes the availability of end-to-end service of the higher-layer protocol. In the network, the whole duration of higher-layer protocol service can be divided into availability time and unavailability time.

The basis for the higher-layer service availability function is thresholds on the SSR and HLSD parameters. The higher-layer service is available on an end-to-end network if the SSR is greater than a certain threshold and the HLSD is lower than a certain threshold. Otherwise, the higher-layer service is unavailable.

The thresholds are only used to make sure that the service provided by higher-layer protocols can be supported by the network, they cannot be used as an index to assess the performance of the service. If the carrier's IP network can provide multi-service, it is better that different thresholds are used for different services.

NOTE – All of the parameters of the higher-layer protocol are abstracted. Each higher-layer protocol may need to make its own instance based on these parameters.

Appendix I

Embodiment example for higher-layer services

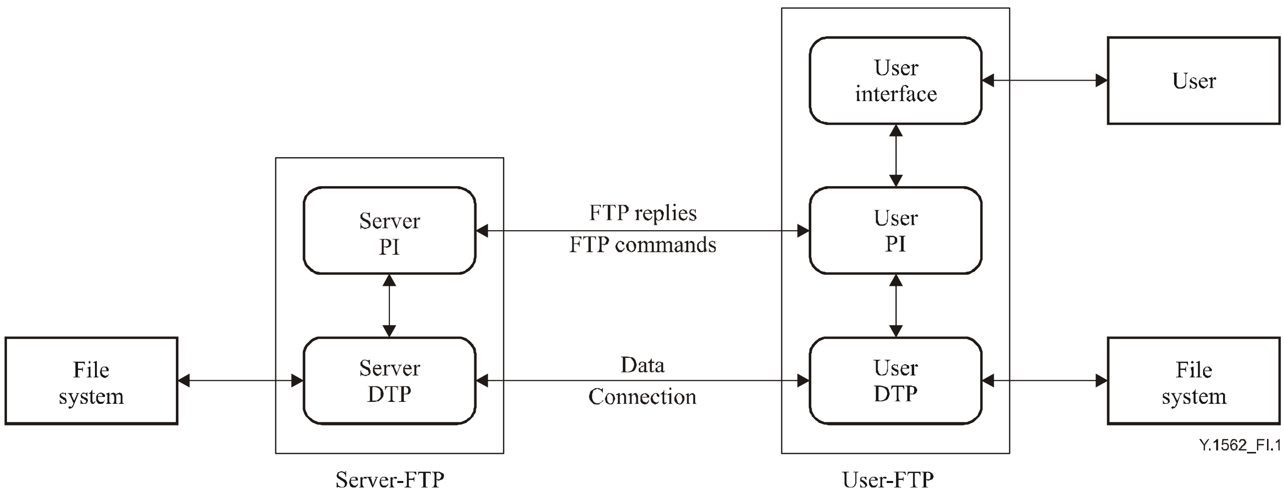
(This appendix does not form an integral part of this Recommendation)

This appendix applies the performance description process in characterizing the performance of a particular higher-layer IP application: FTP file transfer. The FTP protocol is one of the oldest and most widely used protocols in the TCP/IP protocol suite, and its performance is hardly unexplored: in fact, most FTP client applications have some type of performance measurement capability built in. FTP is nevertheless useful as an example NGN application in this study: it is stable, well defined, and familiar, and its QoS characteristics are relatively intuitive. Given its central role in the TCP/IP protocol suite, FTP's functionalities are likely to be similar to those of many higher-layer NGN applications – and it (or a similar protocol) will often underlie them. An analysis of FTP QoS may thus uncover opportunities or difficulties in using the general performance description approach described here in characterizing other, more complex higher-layer IP applications.

NOTE – Other higher-layer services can derive their own reference models and parameters based on this Recommendation following this example.

The FTP protocol model is reproduced in Figure I.1. It comprises four principal entities: a user, user-FTP and server-FTP processes, and a file system associated with each FTP process. The user-FTP process is subdivided into a user protocol interpreter (PI), a user data transfer process (DTP), and a user interface; the server-FTP process is subdivided into a server PI and a server DTP.

FTP commands and replies are communicated between the user and server PIs over a *control channel* that follows the Telnet protocol; user information is transferred between the server and user DTPs over a logically separate *data channel*. Both channels operate over transmission control protocol (TCP) connections. The user controls the user-FTP process via the (non-standardized) user interface. In a typical FTP session, the user-FTP establishes the control channel, "listens" on a specified data port, and issues FTP commands that specify the data transfer parameters and the nature of the file system operation desired (retrieve, store, append, delete, etc.). The server-FTP process carries out the user-FTP commands, establishing temporary bidirectional TCP connections and transferring files as necessary, and responds to each command with one or more server-FTP replies. The DTP interactions with the file systems are not standardized.



NOTE 1 – The data connection may be used in either direction.
NOTE 2 – The data connection need not exist all of the time

Figure I.1 – Model for FTP use (from IETF RFC 959)

Figure I.2 combines the FTP model above with the communication performance model of Figure 6-2. It identifies the two users as a client and a server, and illustrates the associated protocol stacks. In this example the application (FTP) server and the network are regarded as one performing entity, and accordingly there is only one interface at which reference events need to be identified: the UNI. The control and data channels are modelled as one physical connection.[1] The reference events that can be used to define performance parameters in this example are transfers of control information (FTP *commands* and *replies*) and user data packets across the UNI. The elementary *functions* that can be defined on the basis of these reference events are the command/reply pairs (or ordered sets) specified in [b-IETF RFC 959]. Obviously, more complex functions could be defined by concatenating elementary functions.

In FTP transactions, the control connection establishment and close functions are very analogous to the access and disengagement functions defined in traditional communication performance standards, and their performance could be described in a similar way. That task is not undertaken in this contribution, since the establishment of TCP connections is not an FTP (or Telnet) protocol function.[2]

The focus in this example is on the FTP data transfer. [b-IETF RFC 959] defines fifteen data transfer functions, called *file action commands*.[3] Each could, in principle, have performance characteristics of interest to FTP users. This illustrates what is certain to be a major challenge in application performance description: in contrast to traditional communication services, the number of elementary functions supported by IP applications is typically quite large, and they typically can be combined in many ways. In OSI terms, the "hourglass" gets very wide above the transport layer.

One data transfer functions of particular importance to users is addressed in this example: *Retrieve*. This function causes the server-DTP to transfer a copy of the file specified in the *pathname* to the user-DTP. It begins upon issuance of the RETR command at the UNI, and ends (successfully) upon receipt of the corresponding positive final reply (226 or 250) at the same interface, reporting that the requested file transfer has been completed. These *packet exit* and *packet entry* events can be represented by protocol state transitions and can be time-stamped with reference to control packet transfers at the UNI as described in clause 6.

---

[1] In [b-IETF RFC 959], the user-FTP process uses the same IP address and port ("socket") for both connections by default.

[2] Like many other higher-layer protocols, these protocols are *users* of TCP. Parameters describing TCP connection set-up performance have been defined in earlier studies.

[3] They are: Allocate (ALLO), Restart (REST), Store (STOR), Store Unique (STOU), Retrieve (RETR), List (LIST), Name List (NLST), Append (APP), Rename From (RNFR), Rename To (RNTO), Delete (DELE), Remove Directory (RMD), Make Directory (MKD), Print Working Directory (PWD), and Abort (ABOR).

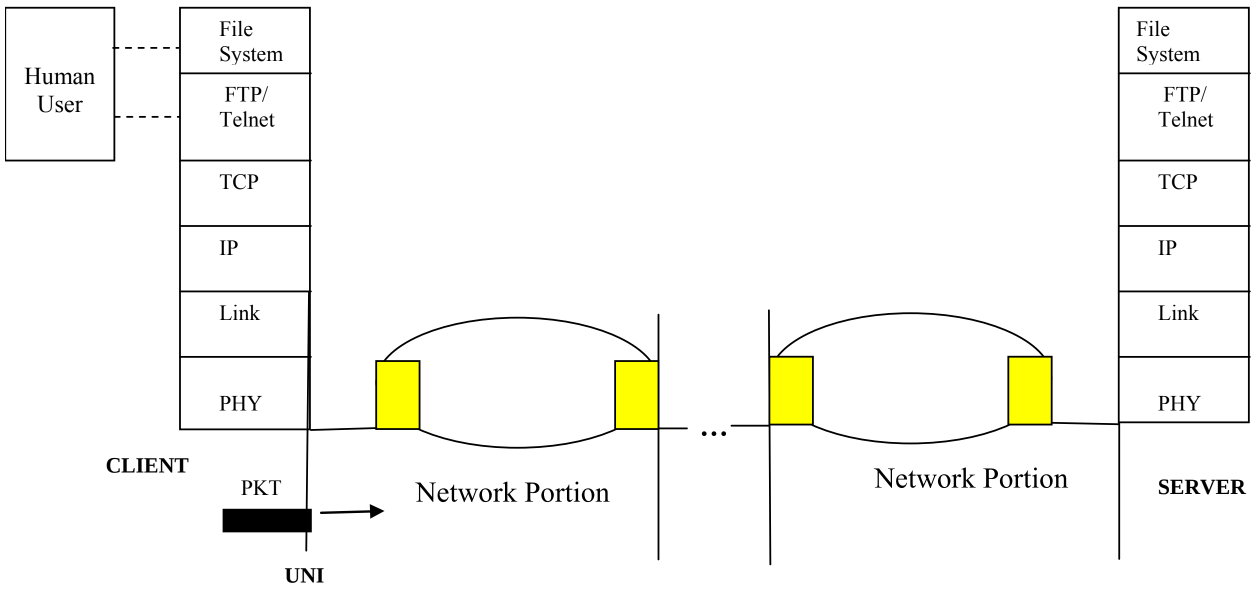


Figure I.2 – FTP performance model

[b-IETF RFC 959] defines a protocol state diagram for the RETR command. This state diagram is reproduced (with minor editorial modifications) in Figure I.3. The state diagram defines three possible *outcomes* of a RETR command sequence: *success*, *error*, and *failure*. These outcomes appear to correspond closely with the generic performance outcomes described earlier. Given that, it is tempting to use the command sequence outcomes directly in defining sample spaces and primary performance parameters for the Retrieve function. However, the RETR outcomes do not distinguish the entity (client or service provider) responsible for the error and failure outcomes. From the response codes defined in [b-IETF RFC 959], it appears that the client is declared to be responsible for most of the error and failure outcomes, and this may well be the case in practice, but servers and communication networks do of course fail and make errors. Some of the response codes clearly report such transgressions.[4] From the standpoint of user-oriented performance description, it is important to define sample spaces and parameters that attribute errors and failures to the entity responsible for them.

In the case of the RETR function, it is possible to eliminate user dependence by assuming that required preliminary activities have been achieved and the input command is valid, i.e.,

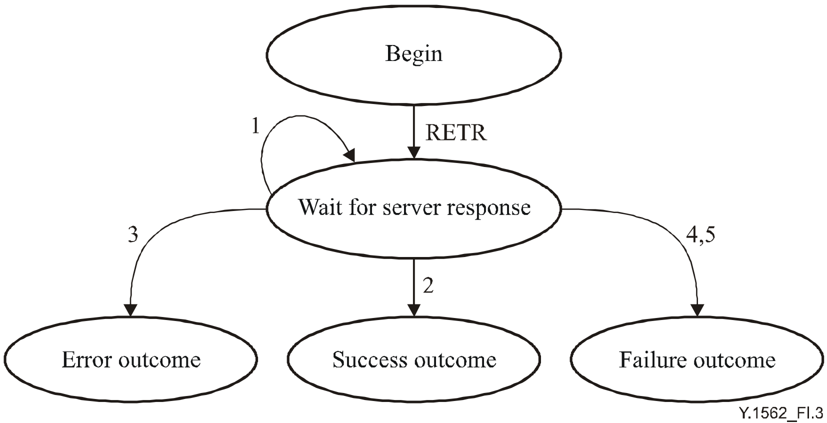
- a control connection has been established, and the user and server data ports have been identified and are ready to exchange traffic;
- data representation formats, transmission modes, and file structures have been specified;
- the RETR command is properly formed, and based on its content it should be possible for the server to perform the requested function (i.e., the referenced directories and requested files exist and should be available at the server).

With these simplifying assumptions, the three performance outcomes for the Retrieve function can be defined as follows:

- **Successful performance**: The correct user data is delivered to the client within the maximum performance time.

[4] Examples: 451 (local error in processing), 421 (service not available), 450 (file unavailable, e.g., busy).

- **Incorrect performance**: Incorrect or incomplete user data is delivered to the client within the maximum performance time.
- **Non-performance**: No user data is delivered to the client within the maximum performance time.[5]



1 Positive preliminary reply
2 Positive completion reply
3 Positive intermediate reply
4 Transient negative completion reply
5 Permanent negative completion reply

Figure I.3 – State diagram for the RETR command (from IETF RFC 959)

As noted earlier, the parameters describing successful performance outcomes are typically relationships of outcomes to time. In the case of the Retrieve function, two obvious parameter choices are *retrieval time* (positive difference between the RETR command and response event times at the UNI) and *retrieval transfer rate* (ratio of total retrieved user data bits to retrieval time). For a given user data file length (which should be specified), the values for these parameters will be affected by the channel transmission rates, transit delays, and error performance of the control and data channels. They will also be affected by the response time of the server-FTP process and its file system. The parameters describing incorrect performance and non-performance are typically ratios of outcomes to trials. Two reasonable parameter choices in this example are *incorrect retrieval probability* and *retrieval failure probability*. The outcome definitions could be formalized using flowcharts and the parameters could be defined mathematically in a rigorous specification.

It would not make sense to define FTP availability performance parameters on the basis of the Retrieve function alone: it is too elementary to characterize an overall FTP service from the general, macroscopic view availability parameters are intended to represent. A specification of FTP service availability would need to be based on an *application profile* encompassing several (or perhaps many) of the elementary FTP functions. Either a time-based or transaction-based approach to availability specification could be used, depending on the characteristics of the defined application profile.

Measurement session

The measurement session of FTP described in this clause is applicable to both IPv4 and IPv6 networks. The procedure includes a full FTP interaction between MN and server. In a single measurement, MN sends measurement request to server to establish control connection. After receiving a request from MN, server will establish control connection of FTP. There is an

[5] Delivery implies that the client is informed of the data's arrival by the expected control response.

authentication procedure in FTP measurement session. Then, data connection can be established between MN and server, and data can be transferred for measurement.

Parameters of FTP service

Based on the performance parameters and availability parameters of the higher-layer protocol, the parameters that can be used to assess the FTP service are as follows:

– FTP Service Response Delay (FTP SRD);
– FTP Service Authentication Delay (FTP SAD);
– FTP Data Transfer Delay (FTP DTD);
– FTP Control Service Delay (FTP CSD);
– FTP Data Service Delay (FTP DSD);
– FTP Control Connection Establishment Success Ratio (FTP CCESR);
– FTP Data Connection Establishment Success Ratio (FTP DCESR);
– FTP Service Success Ratio (FTP SSR);
– FTP Service Availability (FTP SA).

Bibliography

[b-IETF RFC 791] IETF RFC 791 (1981), *Internet Protocol (IP), DARPA Internet program protocol specification*.

[b-IETF RFC 959] IETF RFC 959 (1985), *File transfer protocol*.

[b-IETF RFC 2460] IETF RFC 2460 (1998), *Internet Protocol, Version 6 (IPv6) Specification.*

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
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	Cable networks and 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	Telecommunication management, including TMN and network maintenance
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, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects and next-generation networks**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2007