International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.800

Amendment 1

(03/2009)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Digital networks – General aspects

Unified functional architecture of transport networks

Amendment 1: Techniques to enhance the availability of transport networks

Recommendation ITU-T G.800 (2007) – Amendment 1

ITU-T

International Telecommunication Union

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
General aspects	**G.800–G.809**
Design objectives for digital networks	G.810–G.819
Quality and availability targets	G.820–G.829
Network capabilities and functions	G.830–G.839
SDH network characteristics	G.840–G.849
Management of transport network	G.850–G.859
SDH radio and satellite systems integration	G.860–G.869
Optical transport networks	G.870–G.879
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999

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

Recommendation ITU-T G.800

Unified functional architecture of transport networks

Amendment 1

Techniques to enhance the availability of transport networks

Summary

Amendment 1 to Recommendation ITU-T G.800 includes a new clause describing transport network availability enhancement techniques, the introduction of transport methods over composite links and differentiated connections and a description of complexity and scalability of systems. In addition, it also includes some small changes to existing clauses.

Source

Amendment 1 to Recommendation ITU-T G.800 (2007) was approved on 9 March 2009 by ITU-T Study Group 15 (2009-2012) under Recommendation ITU-T A.8 procedures.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). 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

Recommendation ITU-T G.800

Unified functional architecture of transport networks

Amendment 1

Techniques to enhance the availability of transport networks

1) Clause 3, Definitions

Add the following definitions:

access transport entity: A transport entity responsible for the transfer of information from the access ports at the inputs of a set of termination sources to the access ports at the outputs of a set of termination sinks. The integrity of the information transfer may be monitored. It is formed by combining a set of termination functions and a network transport entity.

communication: A body of information produced by a sender and intended, in its entirety, to reach a particular receiver or set of receivers.

forwarding port: An input or output of a transport entity or layer processor function, input of an adaptation source function or termination sink function, or output of an adaptation sink function or termination source function. The forwarding port on a transport entity is coincident with (corresponds to) a forwarding port on the transport processing function that directly supports that transport entity.

forwarding point: The binding of an output forwarding port and an input forwarding port.

forwarding function: A transport processing function that supports a transport entity in a subnetwork.

layer processor function: A transport processing function that accepts layer network characteristic information at its input forwarding port and delivers layer network characteristic information at its output forwarding port and provides specific transport functionality by reading, modifying or inserting layer information.

link connection: A transport entity that exists within a link that transfers information present at the input forwarding port to the output forwarding port.

network forwarding relationship: A transport entity formed by binding a set of subnetwork transport entities and/or link connections to provide connectivity between a set of forwarding end points.

subnetwork transport entity: A transport entity that exists within a subnetwork (examples of a subnetwork transport entity are subnetwork connection, flow domain fragment, protected subnetwork connection, etc.).

transport entity: An architectural component that exists within a topological component (link, subnetwork or layer network) which transfers information between its input ports and output ports. The information transfer is controlled by forwarding rules. Forwarding between an input port and one or more output ports is controlled by one or more forwarding rules. A transport entity contains one or more forwarding rules.

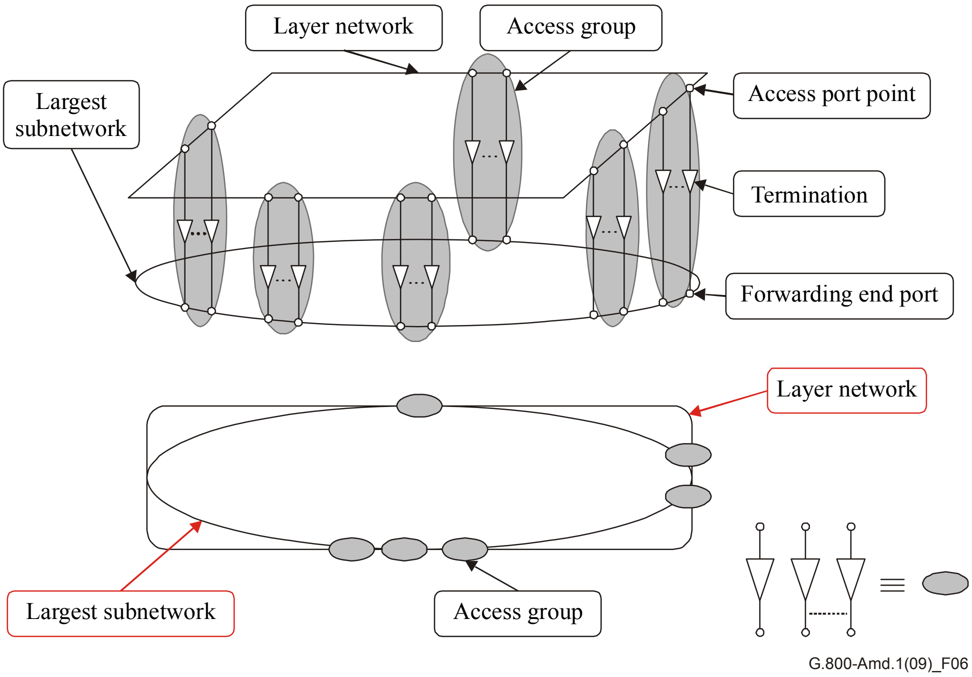
2) Clause 6.3.3

Modify the clause as follows:

A link consists of a link port at the edge of one subnetwork or access group ~~which is associated with~~ and a corresponding link port at the edge of another subnetwork or access group that are associated for the purpose of transferring characteristic information. The link represents the topological relationship and available transport capacity between a pair of subnetworks. Multiple links may exist between any pair of subnetworks.

3) Figure 6

Modify Figure 6 to appear as follows:



4) Clause 6.5

Replace the last sentence in clause 6.5:

The following basic entities are described: forwarding relationship, link connection, connection and transfer association.

with:

The following basic entities are described: forwarding relationship, link connection, connection, differentiated connection and transfer association.

5) New clause 6.5.4

Add the following text as a new clause and renumber existing clauses 6.5.4, 6.5.5 and 6.5.6:

6.5.4 Differentiated connection

A differentiated connection is a transport entity that transfers information belonging to multiple communications between ports across a subnetwork. A differentiated link connection is a special case of a differentiated connection that exists in the context of a link. In a differentiated connection message, contents are interpreted to identify (sets of) communications which receive different treatment. The sets of communications may be distinguished by the forwarding identifier or other layer information. Order is not necessarily preserved between messages belonging to sets of communications receiving different treatment. Sets of communications may be identified for purposes such as traffic conditioning or preserving communication message order.

6) Clause 6.5.4

Renumber to clause 6.5.5 as described above and modify the last two paragraphs as follows:

In a network that uses channel forwarding, the access relationship is supported by a network connection, i.e., it is equivalent to a trail in [ITU-T G.805]. The access relationships provided by such a network are channel based.

In a network that uses destination-based forwarding, the access relationship is supported by a corresponding destination forwarding relationship in the largest subnetwork. The access relationships provided by such a network may be channel or destination based.

7) Clause 6.10

Add the following new subclauses after Figure 16:

6.10 Transport over composite links

6.10.1 Transport over composite link for transport resilience

If a differentiated link connection is configured over a composite link, its ingress can distribute individual communications over component links based on a distribution algorithm. If the differentiated link connection has reserved spare capacity on the composite link, it can redistribute impacted communications to other available component links when a component link fails or is degraded. A distribution and redistribution algorithm can use component link attributes, available information on communications, and policies for distribution decision. As a result, a composite link can be used to gain transport resilience.

6.10.2 Faults and their supervision in a composite link

Component links in a composite link can fail independently, which causes composite link capacity reduction. The scenario is referred to as a composite link constituent fault.

A constituent fault on a composite link can affect its link connections in two distinct ways, depending on the distribution function used by the composite link. If the composite link distributes each link connection it supports to only one component link, then a partial fault may cause some link connections to fail completely and others to be unaffected. The failed link connections may be redistributed over other working (non-failed) component links. If a composite link treats a link connection as a differentiated link connection and distributes communications to different component links, then a partial fault causes the link connection traffic to be redistributed over the remaining working (non-failed) component links. In both cases, the redistribution can reduce the link capacity available to these or other link connections supported by the composite link.

Whether a composite link function must send status information indicating reduced capacity or failure of link connections depends on the recovery mechanisms in use. If the composite link function can fully recover the lost connectivity locally by redistributing traffic across the working (non-failed) component links, it may not be necessary to send status information. If local action is not sufficient to recover lost connectivity, status information may be sent on selected connections depending on recovery policy. For example, some link connections may be selectively shutdown to avoid others being affected or several link connections may signal reduced capacity to their connection endpoints to allow connectivity for all (fair reduction).

8) New clause 10

Add new clause 10 and its associated subclauses as follows:

10 Transport network availability enhancement techniques

In [ITU-T G.805], several transport network availability enhancement techniques are described. This clause describes additional techniques which may be used to enhance the availability of a transport network. These techniques include differentiated connection protection and composite link protection.

10.1 Differentiated connection protection

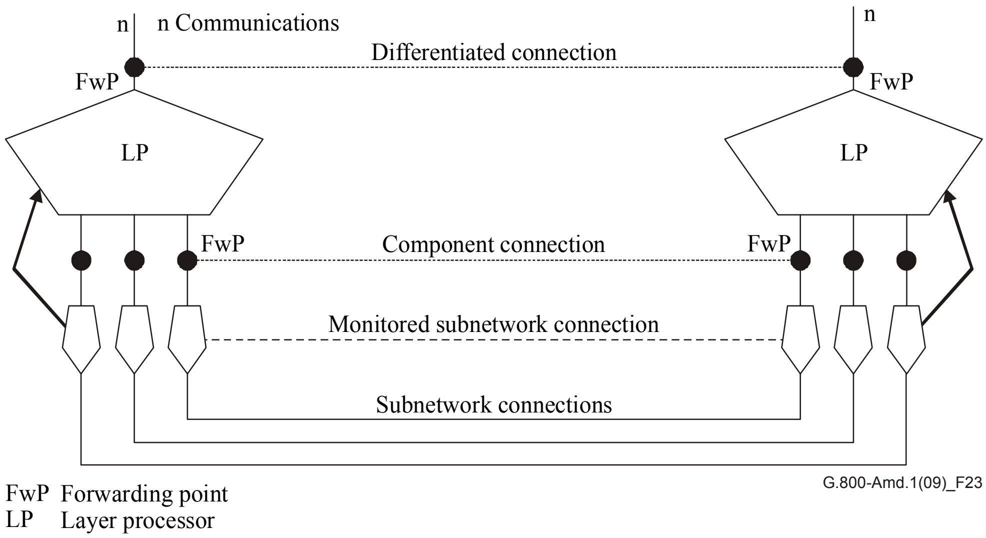


Figure 23 – Transport model for a differentiated connection

A differentiated connection, defined in clause 6.5.4, can provide transport resilience. Figure 23 illustrates a differentiated connection transport model in a layer network. A differentiated connection has several component connections and is configured on layer processors (LP) residing at the differentiated connection ingress and egress. Each component connection is a connection as defined in clause 6.5.3. The component connections can have different capacities. The differentiated connection ingress and egress each have a single forwarding point (FwP). The component connections are subnetwork connections and may be routed through different paths in the layer network. The layer processor at differentiated connection ingress distributes traffic units to the component connections. Each component connection is independently monitored.

The differentiated connection carries multiple communications. The ingress LP can distinguish communications or sets of communications by examining datagram LI contents and distributes each communication to a single component connection. The egress LP takes datagram from each component connection and delivers them to the differentiated connection egress port. Each subnetwork connection will preserve the packet sequence of the communications it carries. The distribution function can consider communication traffic attributes and generate a mapping table between communications and component connections. In subnetwork connection failure situations, the distribution function implements a new distribution relationship. When a failure is detected, the LP implements a new mapping relationship. As a result, the differentiated connection provides resilient transport without using connection protection. Spare capacity must be reserved in the component connections to support this transport resilience mechanism.

A differentiated connection and its component connections may be configured as bidirectional. The two differentiated connection LPs may independently distribute communications. Thus, a bidirectional client communication may be transported over different subnetwork connections in each direction. If bidirectional communications are required to be transported over a single bidirectional subnetwork connection, the two LPs must use the same mapping relationship. In this case, one distribution function provides the mapping policy to both LPs. Each LP executes the provided distribution policy. For more rapid recovery, the distribution function can pre-calculate the different failure scenarios and recovery plans, and provide multiple distribution policies to the LPs. When a component connection fails or is repaired, some information exchange is necessary to ensure both ends use the same distribution policy.

10.2 Composite link protection

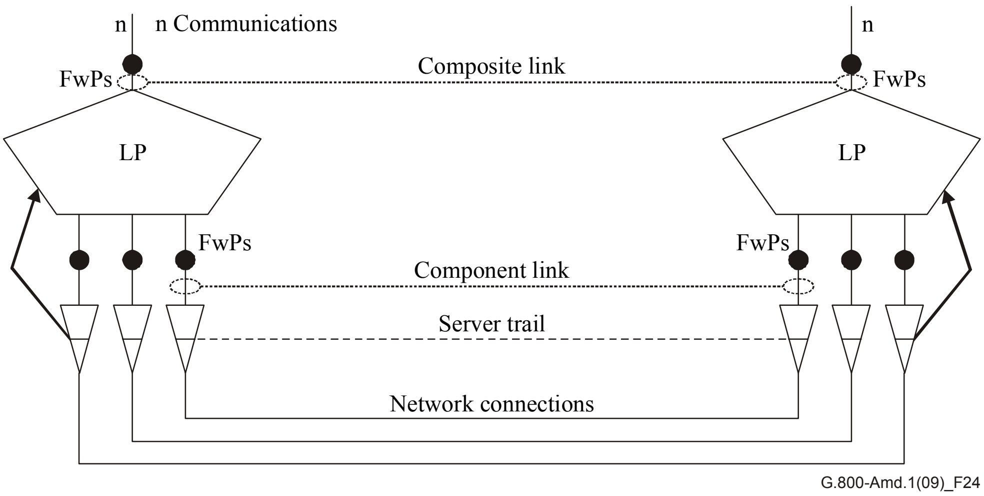


Figure 24 – Transport model for composite link (destination forwarding network)

A composite link defined in clause 6.10 can provide transport resilience without dedicated protection links. The transport model is shown in Figure 24. Multiple component links are bundled together into a single composite link. The component links are supported by independent server trails that are supported by individual server layer network connections. Layer processors reside at the composite link ingress and egress. All packets arriving at the composite link ingress are transported to the composite link egress but order may not be preserved between packets traversing different component links. To preserve packet sequence for individual communications, the distribution function in the layer processor uses LI in the packet to distinguish communications and sends each communication or set of communications over a single component link. For example, a destination address may be used by the distribution algorithm to ensure individual communications traverse a single component link. For finer distribution, other fields may be used as well. A component link failure will trigger the distribution algorithm to change the distribution to use only the remaining active component links. This provides transport resilience. In general, the distribution algorithm may not support traffic engineering due to lack of traffic engineering information for individual communications. However, if the network is aware of communication traffic characteristics, the layer processor can perform the distribution based on this traffic engineering information.

9) New Appendix III

Add a new appendix as follows:

Appendix III

Complexity and scalability of systems

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

Many human endeavours, including engineering, show characteristics of complexity. 'Complexity' has many definitions; however, the definition used in this Recommendation is the following:

- *Complexity* is the ratio between the number of interactions between individuals and total number of those individuals.

If this ratio grows with the number of individuals, then the endeavour can be said to become complex as it gets bigger. This can be seen in many diverse places including business organizations, industry organizations, systems of government administrations, as well as in engineering.

In engineering, the discipline of systems engineering has evolved to deal with complex engineering problems and, in the language of systems engineering, we call these *complex systems*. There are two important properties which characterize many complex systems, with some systems exhibiting one or other characteristic and some exhibiting both characteristics:

- *Chaotic behaviour* – Complex systems can have unexpected, emergent behaviour that often runs counter to the basic objectives of good engineering where predictable behaviour is essential – we expect that the response of a bridge to a cross wind has been correctly predicted when we drive across the bridge.
- *Non-linear cost of expansion* – In a complex system, when there is an inherent cost of interactions between individuals, the costs of the overall system can grow more than linearly as the system is expanded.

Generally speaking, complexity is an undesirable characteristic for a system. The discipline of systems engineering largely deals with the good engineering practice for managing the development of systems which have a given and unavoidable level of complexity.

However, the complexity of a system is often a matter of choice. One particular system architecture may be highly cost effective at a small scale; however, if the system needs to be expanded, its inherent complexity means that costs will increase dramatically with the expansion. An alternative system architecture may be more expensive at small scale; however, because it has less inherent complexity, the increase in costs when it is expanded is much lower.

Although chaotic behaviour can be beneficial in a very few systems; in the engineering of the great majority of systems, it is highly undesirable. At minimum, chaotic behaviour can increase the overall costs as the final costs of mitigating the unexpected behaviour, many of which may not have need expected at the design stage, must be added to the overall system. A minimum consequence of chaotic behaviour is to increase the non-linearity of the cost of expansion.

Frequently, the practical consequence of complexity within a system architecture is the scalability of the system. The scalability can be usefully defined as the following:

- The *scalability* of a system is the power exponent of the total costs as a function of expansion. Generally speaking, a power exponent of 1 (or suitably close to 1) is called *scalable* while an exponent significantly greater than 1 is said to be *non-scalable*. The *scaling limit* of a system is said to be the scale at which the exponent becomes significantly greater than 1.

Achieving an architecture for a large system which is scalable and which does not have a clear scaling limit is normally not straightforward. Generally, scalable systems arise when close attention is paid to the way in which interactions grow as the overall system grows.

III.1 Independence of subsystems

In a system comprised of subsystems, complexity arises out of the interactions between subsystems. If we follow the consequences of a scalable system, we can see that:

- When a new subsystem is added to a system, for the system to be scalable, the number of interactions generated by the new subsystem must be fixed with the new subsystem and not depend on the total number of subsystems in the overall system.

This observation leads to an analysis of dependencies between subsystems. A subsystem has a dependency on another subsystem if it has any interaction with it. Analysis of dependency is one of the features built into modelling languages, notably the unified modelling language (UML).

Once a dependency has been identified, the nature and frequency of interaction can be categorized. Examples include the following:

- A dependency between subsystems may exist only to manage the process of adding and deleting a subsystem.
- A dependency between subsystems may exist for occasional ad hoc interactions; for example, the dependency between an individual web browser and an individual web server.
- A dependency between subsystems may require time critical, high volume, state locked interactions.

However, a simple conclusion of scalability is that for a large system to be scalable, most subsystems must work independently of each other. Maximizing independence between subsystems is at the heart of scalability.

III.2 Independence within a lifecycle and between lifecycles

The engineering of a system involves a lifecycle which broadly comprises – requirements capture, architectural design, component design, component development, deployment, live operation, operational support and repair, decommissioning.

The discipline of systems engineering has given considerable attention to managing the dependencies between these different stages in the engineering lifecycle. This is generally a *forward dependency*. For example, live operation and operational support and repair may well depend on decisions made in design stages. In this case, the observation has been that complexity has been generated by assuming perfection in the early stages with an assumption that there are no forward dependencies but in practice, this turned out not be the case. The development process associated with this is often referred to as the 'waterfall' method. In practice, some dependency is largely unavoidable so complexity is managed better by acknowledging and planning for the dependencies. This has led to the current systems engineering development process of 'iterative' cycles.

Importantly for telecommunications, the live operation phase cannot be readily 'turned off'. This means when a new system is introduced, it must be integrated into existing systems – the old and the new form one large super system. This means that there is now not merely dependencies between the stages of a particular development lifecycle, but there are dependencies between different lifecycles. This can take many forms. Examples include the following:

- Between the architectural design stages of each lifecycle, there are a new set of potential subsystem dependencies between the subsystems of each development.
- The live operation of the existing system has a dependency on the architectural design of the new system. The design of the new system may well affect the level of operational disruption caused to the existing system when the new system is deployed.
- There may be a dependency between the deployment of the new system and the decommissioning of the old system.

III.3 Transparency of telecommunications network services

Telecommunications systems introduce a further aspect of dependency. Broadly speaking, this is "when a client uses a server layer network, is there a dependency between client and server?" We can consider this in two stages: first, clarify what is meant by dependency and what is independence in the specific case of telecommunications; secondly, examine specific examples to show the extent and characteristics of dependency between client and server.

When a client uses a network to transfer information between end points, as set out in the axioms of ITU-T G.800, there is one basic independency and five basic forms of dependency setup.

- Symbol selection independency – this is the extent to which the client can select symbols from a lexicon at will without creating any dependency interaction in the server. This is defined as the *transparency* of the server.
- Lexicon dependency – this is the delineation of traffic units which the client wishes to transfer. The client has full freedom and independence in the selection from the lexicon but the lexicon itself and the demarcation between sequenced symbols from the lexicon are a dependency between client and server.
- Attachment control information dependency – this is the interaction between client and server to establish an access point to the server network and assign it an address which is shared information.
- Communication control information dependency – this is the interaction between client and server requesting a particular transfer of information between access points. This normally involves the client passing the server a set of destination addresses and a source address which may be implicit by the location of the request.
- Performance control information dependency – this is the interaction between client and server indicating the performance requirement and its scope may vary, for example, the scope may be a communication or it may be a symbol/traffic unit.
- Transfer performance dependency – the success of the communication cannot be totally guaranteed and so there is a performance dependency setup between client and server.

Having established these dependencies, it is possible to characterize different forms of server network according to the characteristics of these dependencies. Some examples are illustrated in the table below.

Table III.1 – Examples of dependency for server layer networks

	Independency	Dependency				
	Transparent symbol selection	Lexicon and symbol demark	Attachment control information	Communication control information	Performance control information	Transfer performance
Fixed rate leased line	Binary bit	Unit interval	At 'subscription' time	At 'subscription' time	None	EP, AP, SR
Packet PVC with uniform diffserv model	Packet payload	Frame length field	At 'subscription' time	At 'subscription' time	Code point with every packet	EP, PL, AP, SR
PSTN	Analogue amplitude/ time quanta	Real time	At 'subscription' time	With every connection request	None	Noise, AP
Public Internet	Packet payload	Frame length field	At 'subscription' time (may be PPP over server connection)	Destination address with every packet	None	EP, PL, AP
Private IPVPN service with uniform diffserv model	Packet payload	Frame length field	'Subscription' time interaction per end point of VPN	Destination address with every packet	Code point with every packet	EP, PL, AP, SR
EP – error performance; PL – packet loss; AP – availability performance; SR – shared risk of common mode failure between sets of communications.						

As can be seen, the number and rate of interactions associated with the dependencies vary greatly. The choice of level of interaction for any client/server relationship will inevitably affect the overall scalability and complexity of a telecommunications network.

It is also possible to include further dependencies. The server can use further information from the client, for example to control routing choices. However, three factors must be considered when constructing further dependencies between client and server layers.

- This inevitability increases the basic complexity of the overall telecommunication network and may well fundamentally undermine scalability.
- This is likely to generate an inter-lifecycle dependency between the development lifecycle of the client and all its possible servers as well as between the server and all its possible clients. This may have profound consequences for the complexity of evolution for the overall network.
- If the dependency is based on symbol selection over which the client thought it had full and independent choice, then the introduction of this feature fundamentally reduces the transparency offered by the server to the client.

10) New Bibliography

Add Bibliography as follows:

Bibliography

[b-Shannon] Shannon, Claude E. (1948), *A mathematical theory of communication*, Bell System Technical Journal, 27 pp. 379-423 and 623-656, July and October.

[b-Trudeau] Trudeau, R.J. (2004), *Introduction to Graph Theory*, Dover Publications.

[b-Watanabe] Watanabe, O. (1992), *Kolmogorov Complexity and Computational Complexity*, Springer Verlach, October.

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	Terminals and subjective and objective assessment methods
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, 2009