SAE/AS-2A SUBCOMMITTEE
RTMT*
STATEMENT ON REQUIREMENTS FOR
REAL-TIME COMMUNICATION PROTOCOLS (RTCP)
* REAL-TIME MODEL
TASK GROUP

ISSUE #1
(Final Draft)

November 12, 1991

SAE ARD50007

Contents

Figures

ABSTRACT

This document contains the requirements for the SAE/AS-2A Real-Time Communication Protocols (RTCP) as developed by the RTMT Task Group. These requirements are an edited compilation of those voiced by numerous military and commercial organizations primarily concerning aircraft applications. However, it is the opinion of the Task Group that these requirements have general applicability to distributed real-time control systems.

1.0 PREFACE

1.1 ADMINISTRATIVE

This document was prepared under the direction of the following administrative personnel of the SAE Avionics Systems Division, to whom general inquiries and/or recommendations should be directed:

Vernon Nicklow, Chairman AS-2 Committee
c/o TRW
13567 Stoney Creek Road
San Diego, CA 92129 619/592-3431

Christina Berggren, Chairman AS-2A Subcommittee,
c/o International Business Machines Corporation
P. O. Box 6
Endicott, NY 13760 607/755-0123
berggren@gdls4.vnet.ibm.com

1.1.1 CHARTER

The Real-Time Model Task Group was chartered by the Systems, Applications, and Requirements Subcommittee (AS-2A) as follows :

Define requirements for communication protocols specifically intended for real-time applications. These requirements are intended to complement SAE AS-2 communication standards, and address the logical interface between user and communication tasks, and the functions allocated to communication tasks.

Systems for real-time applications are characterized by the presence of hard deadlines where failure to meet a deadline must be considered a system fault.

1.2 FOREWORD

A wide spectrum of industry representation has been tasked to define the requirements for a multinational, multiservice, commercially accepted military standard for real-time communication protocols. This effort was sponsored by the Society of Automotive Engineers (SAE) Interconnect Networks committee AS-2. Inputs to this document have been consolidated from participating resources of the Department of Defense′s military services, aerospace prime contractors, United

Kingdom, France, West Germany, electronic and LSI component suppliers, ARINC, commercial aviation equipment manufacturers and consumers, and systems consultants. The requirements defined herein discuss technical aspects based upon the following considerations:

- Functions
- Performance
- Test Requirements
- Operational Properties
- Validation Requirements

Comments and recommendations for changes to this document are solicited and should be addressed to the following:

Christina Berggren, Chairman AS-2A-2 RTMT Task Group
c/o International Business Machines Corporation
P. O. Box 6
Endicott, NY 13760 607/755-0123
berggren@gdls4.vnet.ibm.com

Guy d'Hervilly, Secretary AS-2A-2 RTMT Task Group
Dassault Electronique
55 Quai Marcel Dassault
92214 Saint Cloud
France 33.1.3481.4505

1.3 NOTICE

This vesion of the document is the final draft. It is ready to be submitted to SAE for publication. Committee approval was obtained by mail ballot completed September 20, 1991. The changes that have been incorporated are editorial or explanatory in nature.

2.0 SCOPE

The purpose of this document is to establish the requirements for Real-Time Communication Protocols (RTCP). Systems for real-time applications are characterized by the presence of hard deadlines where failure to meet a deadline must be considered a system fault.

These requirements have been driven predominantly, but not exclusively, by aerospace type military platforms and commercial aircraft. These requirements are primarily targeted for the Transport and Network Layers of peer to peer protocols, as referenced in the *Open System Interconnect Reference Model (OSI/RM)*, developed by the International Standards Organization (ISO). These requirements are intended to complement SAE standards *AS4074.1 (High Speed Linear Bus Standard)* and *AS4074.2 (High Speed Ring Bus Standard)*, and future SAE communications standards. Although information transfer objectives herein concentrate primarily on digital data flow attributes, efforts have been made such as not to preclude sensor/video and voice information transfers. This document provides evaluation criteria for selecting viable alternatives for a real-time communications protocol standard.

3.0 INTRODUCTION

The RTCP requirements set forth in this document, reflect a consensus of experts′ views for satisfying the needs of applications encompassing the entire field of aerospace military platforms and commercial aircraft. Non-aerospace applications are not precluded and will be encouraged to utilize the proposed RTCP standard. Aerospace categories of the RTCP intended applications are outlined in Figure 1.

MILITARY PLATFORMS	CIVILIAN AIRCRAFT
o Space Craft	
- Manned	
- Unmanned	
o Manned Aircraft	o Manned Aircraft
- Fixed Wing	- Fixed Wing
Command Post	Transport
Bomber	
Transport	
Fighter	
Anti-Sub Warfare	
Ground Attack	
Airborne Early Warning	
- Rotary Wing	- Rotary Wing
Light	
Heavy	
o Unmanned Aircraft	
- Missile	
- Remote Piloted Vehicle	

NOTE: Multiservice
Multinational

Figure 1. Aerospace Categories of RTCP Intended Applications

3.1 TARGET APPLICATIONS DESCRIPTIONS

The protocol requirements defined in this document are intended for real-time applications. Real-time applications, as defined here, are characterized by the presence of hard deadlines where failure to meet a deadline must be considered a system fault. While these applications are open in that they use published standards to insure interoperability between equipments and across platforms, they are closed in the sense that valid users and transactions are predetermined during system design for a specific implementation.

The following description and characteristics of the intended target applications are based on a set of KEY DESCRIPTOR WORDS. These target application key descriptor words were developed by the Real-Time Model Task Group (RTMT) at early meetings, and they represent the consensus of this group. These key words and functional breakdown are included as Appendix A to this document.

3.1.1 DISTRIBUTED

The RTMT target applications are distributed. This means that in addition to applications and processing being distributed, the following are also distributed: data, communications functions, control, storage, and system management functions.

The characteristics of distributed data are manyfold and include the following:

- Demand driven; that is, the data are sent when requested.
- Supply driven; that is, the data are sent at the time that the data are originated.
- For some data categories all data are essential, whilst for other categories only the latest data (or the latest N sets of data) are essential.
- For some data categories the integrity of the data is essential, whilst for other categories integrity is not a driving requirement.
- For some categories of data, the order of arrival, or knowledge of the order of origination, is essential.
- Distributed data are of any size and of different data types (objects).

The characteristics of the communication functions being distributed include the data transfer itself, packetization or the segmentation and reassembly of data, flow control and the routing to target destination(s) of the data.

Characteristics of the control being distributed in these RTMT target applications include the following:

- Application control; that is, control of the initiation of execution of the tasks that are subsets of an application that result from the distribution of said application.
- Provisions for the distribution of a global reference point, for synchronization purposes, and to allow time stamps on data, should that be desired.
- Provisions for remote interrupts.

The characteristics of storage being distributed implies that the target applications facilitate parallel data searches and stores, and the need for guaranteeing data coherence; that is, the data stored must at any point in time appear consistent to all users.

The implications of system management functions being distributed in these RTMT target applications include the following:

- Configurations; both static and dynamic.
- Monitoring of system status, health and errors; the latter being either hard or intermittent.
- System state support; the global understanding of current system state, for instance, normal or exception states with degraded modes and changing system priorities and objectives.

3.1.2 UNINTERRUPTED OPERATIONAL PERFORMANCE

The RTMT intended applications are characterized by requiring uninterrupted operational performance. This uninterrupted operational performance implies the tolerance of the occurrence of both scheduled and unscheduled events. Scheduled events may be executed either off or on line.

The unscheduled events may be either planned, such as interrupts or asynchronous events, or unplanned. The unscheduled unplanned events include both faults and defects.

3.1.3 REAL-TIME

The RTMT intended applications are real-time systems. In this context real-time means that the systems are characterized by the presence of hard deadlines where failure to meet a deadline must be considered a system fault.

Real-time is a relative concept; timing parameters may vary from system to system. Requirements for each application specify the necessary deadlines, i.e., the acceptable response times.

3.1.4 HIGHLY AUTOMATED

The RTMT intended applications are characterized by many functions being automated. To be automated a function must be predefinable, that is, either via analytical methods or parametrically through behavioral models.

3.1.5 CONTROL

The RTMT intended applications may to a large extent be characterized by executing control functions. There are cause/reaction modes of system behavior. This implies closed loop(s) with cyclic data, and a fail safe operation with high integrity. In addition to closed loops the system may contain open loops.

3.1.6 PREDEFINED SYSTEM CONFIGURATIONS

All RTMT intended applications are characterized by having system configurations that are predefined or planned in advance during the system design. Therefore, all system parameters, such as processor or communication network loads, can be calculated and resources allocated. This includes the system dependability as well as the event of changing system states with anomalies resulting in the exception handling of, for instance, data, applications/processing, system management, and initialization/reinitialization.

3.2 SYSTEM DEFINITION

The RTMT intended target applications as described above under paragraph 3.1, “TARGET APPLICATIONS DESCRIPTIONS” can be functionally partitioned into the USER APPLICATION and supporting communications functions, the REAL-TIME COMMUNICATION PROTOCOLS (RTCP) with a logical interface between the two functional areas as shown in Figure 2.

This document defines the requirements for the communications functions that can support target applications as described above. It is understood that there may be a USER SPECIFIC COMMUNICATIONS FUNCTION integrated with the USER APPLICATION. Requirements for this integrated user communications function are not included in this requirements document.

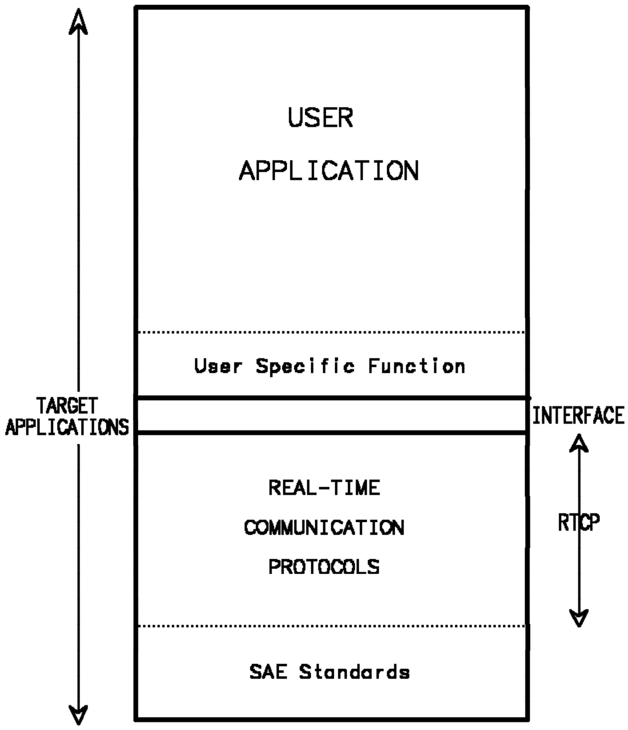


Figure 2. RTCP System Definition

Note: NOMENCLATURE. These target applications are distributed over a set of hardware equipments executing Application and Communication functions. In addition, the Communication function may be allocated to a Network Interface Unit (NIU); that is, when the logical interface shown in Figure 2 maps onto a physical interface between the communications and applications processors.

4.0 REQUIREMENTS

4.1 GENERAL

4.1.1 TRANSFER TOPOLOGIES

Transfer topologies as defined below apply to all system services, that is, to the data communication service (4.2.1), the synchronization service (4.2.2), and to management services (4.2.3).

Rationale:

The transfer topologies described below for User to User(s) transfers are also applicable within the RTCP.

From the theoretical point of view, only unidirectional point-to-point transfers are necessary. However for the real-time applications being considered here, other topologies are useful. Please refer to Figure 3 for definitions of topology terms as used here.

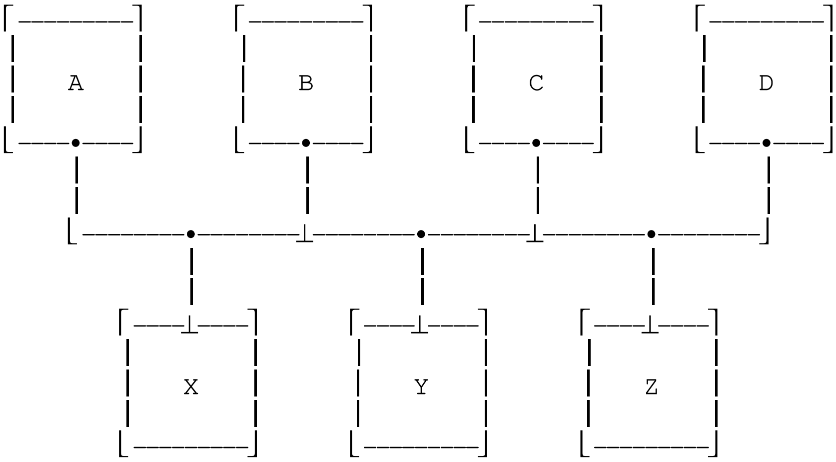
4.1.1.1 Point-To-Point

The RTCP shall support unidirectional point-to-point transfers between any given pair of User entities.

It is desired the RTCP support bidirectional point-to-point transfers between any given pair of User entities.

Rationale:

The unidirectional point-to-point transfer provides the fundamental service. Sometimes, transfers are simultaneously required in both directions between two User entities. Whilst this could simply be provided by two, independent, unidirectional transfers, it would be advantageous for the two transfers to be invoked by a single request. Since the bidirectional link may have to be implemented as two unidirectional links, the advantage is slight and the capability only desired.



POINT-TO-POINT: A transfer from any one station to any other one station; for instance, from A to X, or from B to Y.

BROADCAST: A transfer from any one station to all other stations; for instance, from A to B and C and D and X and Y and Z.

MULTICAST: A transfer from any one station to a number, but not all, other stations; for instance, from A to X and Y and Z, or from B to A and D.

CONCENTRATION: The transfer to any one station from a number of stations with the merger of the different pieces of information into a large concatenated piece of information before presentation to receiving entity; for instance, X receiving data from A, B, and C concatenated into one piece of information "ABC".

MULTIPLE CONCENTRATION: The transfer to more than one station from a number of stations with the merger of the different pieces of information into a large concatenated piece of information before presentation to receiving entity; for instance, X and Y receiving data from A, B, and C concatenated into one piece of information "ABC".

Figure 3. Definition of Topology Terms

4.1.1.2 Broadcast

The RTCP shall support the one-to-all transfers (broadcast), always unidirectional, between any given User entity (acting as the unique source of the transfer) and all other User entities (acting as simultaneous data sinks of the transfer).

Rationale:

For real-time applications, certain information, created at a given source, may be needed by all sinks. If the RTCP offers only point-to-point transfers, this operation would waste a lot of bandwidth since the same information would have to be transmitted (on the same media) many times, repeated for each user. Therefore, it seems reasonable for the RTCP to offer a special type of logical transfer topology which will use a unique transmission (of one message only), but simultaneously destined to all the User entities.

4.1.1.3 Multicast

The RTCP shall support the one-to-many transfers (multicast), always unidirectional, between any given User entity (acting as the unique source of the transfer) and a number of (not all) other User entities (acting as simultaneous and defined targets for the transfer).

Rationale:

For real-time applications, certain information, created at a given source, may be needed by some, but not all, sinks. The rationale is very similar to that for broadcast above. If the RTCP offers only point-to-point transfers, this operation would waste bandwidth since the same information would have to be transmitted (on the same media) several times, repeated for each User that requires the information. Therefore, it seems reasonable for the RTCP to offer logical transfer topologies which will allow transfer of information to groups of Users simultaneously.

4.1.1.4 Concentration

The RTCP shall support the many-to-one transfers (concentration), unidirectional only, between a number of defined User entities (acting as simultaneous, or successive, sources of the transfer) and any other given User entity (acting as the single sink of the transfer).

Rationale:

This is the exact opposite from broadcast/multicast. A key User in the system may need, at more or less regular intervals of time, several pieces of information, coming from different sources. These pieces of information are merged into a single large piece of information and then provided to the User.

That sink (i.e. the RTCP collecting entity) will then be involved in the job of arranging together all the incoming pieces of information in a neat conventional package (ordered and formatted properly) before passing it to its user (the sink host). This frees the host from unnecessary protocol processing.

4.1.1.5 Multiple Concentration

It is desired that the RTCP support the many-to-many transfers (multiple concentration), unidirectional only, between any given pair of groups of defined User entities.

Rationale:

This rationale is as for Concentration. The only difference is that there is now more than one sink for the data, allowing redundancy, for instance.

4.1.2 SEMANTIC IMPORTANCE

The Real-Time Communication Protocols (RTCP) shall be required to provide System Services for different categories of message classes distinguished by their degree of semantic importance.

Semantic importance here is defined as a measure of relative value of completion of a process associated with the transfer. Therefore, a measure of relative value, i.e., semantic importance, shall be associated with the completion of each requested data transfer.

The assignment of semantic importance to a message shall be the responsibility of the system designer.

Rationale:

Semantic importance class assignments for data transfers in RTCP provide the means for support of varying system states. Semantic importance should not be confused with priority; priority provides the means for scheduling. Semantic importance may have no relevance under fault-free conditions; it is only used to determine the action to be taken when particular system faults occur. Two such examples are load shedding and missed deadlines. Semantic importance can also be used to dynamically reschedule mission critical transfers.

If the available system resources are reduced below the minimum required to perform the defined functionality, some of the less important functions will not be performed (load shedding). The transfers associated with these functions will also be shed. This may be indicated by a simple semantic importance number which is compared with the system state to determine whether or not the transfer should take place.

If a transfer misses its deadline the system has failed. However, the information may still be of value such that the recovery action could be to complete the transfer as soon as possible. Alternatively, if the completion of a transfer has no value, it may be desirable to cancel the transfer.

The value of completion of a transfer, i.e., semantic importance, may be a complex function varying with time.

The term "semantic importance" is used here because there have been previous references in the literature using this term to convey the meaning defined above. The RTMT, while recognizing that the word "semantic" does not imply that the content and meaning of related messages are known to the RTCP, elected to use this term to avoid a proliferation of terms having similar meanings.

The task group has not reached consensus regarding the number of classes of semantic importance supported by the RTCP or the components of the relative value classification. Further definition will be included in "Logical Interface Definition for Real-Time Communication Protocols (RTCP)", SAE ARD50033.

4.1.3 PERFORMANCE REQUIREMENTS

The RTCP shall be fully deterministic; that is, its exact behavior can be predicted for any given set of conditions.

In order to meet deadlines, the RTCP shall guarantee the completion of requested services having different latency requirements.

All aspects of the RTCP shall be schedulable.

Rationale:

Since "real-time" is a relative concept, it is not feasible to specify required performance numbers for the RTCP, either in the form of delay or bandwidth from User to User. These will vary from system to system, dependent on the application. The applications envisioned may have latency requirements in the order of hundreds of microseconds. However, it is necessary that the RTCP be predictable in order to guarantee its operation in all circumstances.

It is also necessary to influence the order in which service requests are processed; requests with lower latency requirements should be preferred over those with higher latency requirements. Alternatively, it is possible to define the order in which service requests are to be processed; higher priority requests must be serviced before those of lower priority. If the RTCP protocol processing was implemented as a FIFO with requests serviced in the order they were received, high priority messages may be stuck *behind one of a lower priority, or a low latency message behind those with less strict requirements. Thus, prioritized access at the media level can be completely defeated if a time critical message sits in a queue behind a message that cannot gain access to the transmission medium.*

4.1.4 AUTOMATIC FUNCTION ALLOCATION

The RTCP shall support the allocation to RTCP processing of certain automatic functions as instructed by the User. The RTCP shall have the capability to execute these functions without specific interaction from the User for each initiation, or, at the time of the triggering event.

Rationale:

For some of the transfers supported by the RTCP, it does not make sense to involve the User; the transfers are simple enough that they can be handled by the RTCP automatically once they have been set up. In addition, the RTCP may be able to handle them more efficiently. These automatic functions may include the periodic transmission of synchronization and timing information or messages communicating the current health of the system. Whilst most of these are likely to be in the form of periodic messages, it is anticipated that some may be triggered by specific events; for instance, if so-and-so happens send an alert.

4.1.5 RTCP LAYERS

For purposes of developing implementations, the RTCP shall be considered as a single functional whole, rather than as an assemblage of interfacing layers.

Rationale:

The ISO OSI/RM defines a functional model of a communications system. However, this has been used to define an implementation model, with specific protocols for each layer and real interfaces between each with corresponding performance penalties. To prevent the same problem from occurring with the RTCP, the functional model should be as simple as possible, to ensure the implementations of the RTCP are considered as an integrated whole, not a number of disjoint bits and pieces.

4.1.6 RTCP IMPLEMENTATION

The RTCP definition shall not prevent its implementation in either hardware or software.

Rationale:

For performance reasons, part of the RTCP will most likely be implemented in hardware. However, for implementations that do not have tight performance requirements, the implementation through software may be more cost effective.

4.2 SYSTEM SERVICES

4.2.1 DATA COMMUNICATION SERVICES

Data Communication Services shall provide the means necessary to transmit Data Units between User entities.

The Real-Time Communication Protocols (RTCP) shall transmit the Data Units in a transparent fashion between these User entities. Its role is to free the User entities from having to supply transfer execution details. In particular, the RTCP shall ensure complete management of the transmission of Data Units.

The real-time communications system shall provide the User with two forms of data transfer services:

Category 1: The parameters for a series of transfers are given before any data is passed and thereafter the User provides data only.

Category 2: The data and parameters for the transfer are provided by the User for each transfer.

Rationale:

From the viewpoint of a real-time communications system User, all data transfers can be considered to be connections with other Users, since resources will have to be allocated and the Users must cooperate whether implicitly or explicitly to transfer the data. The way in which the User chooses to control these data transfers, however, results in two sets of differing requirements for services offered by the real-time communications system and covered by these Real-Time Communication Protocols requirements. These two sets of services may be simply (but not definitively) differentiated by the Category 1 transfers having resources specifically assigned to each such that the User provides, or is provided, only data for a transfer, while the Category 2 transfers share some allocated resources such that the User must provide, or be provided with, data and transfer control information.

4.2.1.1 Category 1 Transfers

The RTCP shall provide Category 1 transfer services. These connection oriented transfers are characterized by having specific resources allocated in the communicating peer entities. Some consequences are:

a) Agreement to partake in the exchange of data is needed; there exists an effective cooperation between communicating entities.

b) No target address is required per message. Knowledge of the communicating entities belong to the ″Connection″.

c) The connection has to be built with all its parameters before being used.

Two main classes of Category 1 transfers can be considered:

1. Static: These are characterized by parameter values which do not vary for any invocation. These Category 1 transfers may be in the Non-Existent, Dormant or Active state (see Figure 4).

2. Dynamic: These are characterized by parameter values which may vary between invocations. These Category 1 transfers may be in the Non-Existent or Active status (see Figure 5 on page 18).

A Category 1 transfer is Non-Existent when the parameter values are not explicitly known by the RTCP. In the Dormant state, the parameter values have been loaded and distributed through the RTCP, but all resources have not been allocated. In the Active state, all resources have been allocated and transfers may take place.

Both classes may be invoked for the life of the system or only for short periods during the system life.

Rationale:

A good way to illustrate a Category 1 transfer is a telephone call. The request for a transfer is set up by dialing the number. Once the recipient answers and agrees to converse, no further addressing information is required; you simply talk into the telephone. Resources are allocated by the telephone company to ensure your connection (you hope).

When a particular Category 1 transfer is not required, it is beneficial to free the resources allocated to enable their use by other transfers. If the parameter values are fixed (Static), then some of the resources could remain allocated while others are utilized elsewhere (Dormant). Quick re-allocation from the dormant state can be achieved without resending the parameter values. If the parameter values may vary

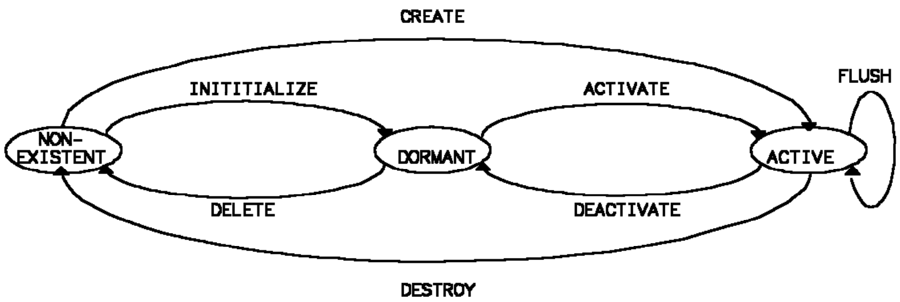


Figure 4. Static Connections State Diagram

(Dynamic), then all resources must be de-allocated as different values may be used next time, requiring the parameter values to be sent again.

4.2.1.2 Category 2 Transfers

The RTCP shall provide Category 2 data transfer services. These connectionless transfers are characterized by having no specific resources allocated. Some consequences of this are immediate:

a) No agreement to partake in the exchange of data is made; there is a generic cooperation.

b) A target address is required per message.

c) All parameters required are supplied per message.

d) Resources are shared in the User Modules allowing diversity.

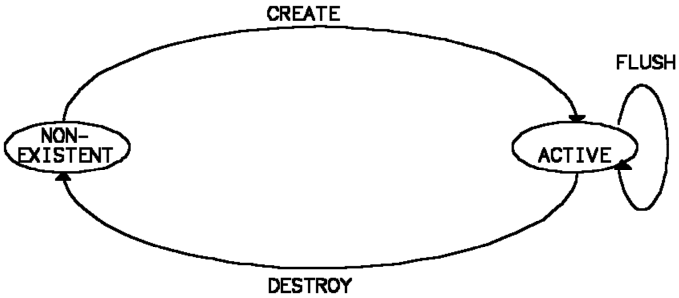


Figure 5. Dynamic Connections State Diagram

Rationale:

A good way to illustrate a Category 2 data transfer is the mailing of a letter. The transfer is initiated by addressing the letter and putting it in the mailbox. This is (hopefully) transferred to the correct person who will (hopefully) read it. Nothing is known of the letter's progress unless specific instructions are included (e.g. registered delivery, RSVP).

It should be noted that, because the target applications of the real-time communications system are closed, the Category 2 transfers are designed into the system. Thus the probability of the transfers reaching the required User(s) may be 1, the same as for the Category 1 data transfers. Of course, to limit the resources required, these transfers may still be designed to reach the required User(s), with a probability of less than 1, if this is acceptable to the system.

4.2.1.3 Message Size

The real-time communications system shall allow the User to define a parameter for a Category 1 transfer defining the size of the message to be passed. It is desired that the bounds of this parameter be unrestricted. A System Design parameter shall also be defined restricting the maximum message size of a Category 2 transfer.

Rationale:

The message size parameters may be applied to either Category 1 or Category 2 transfers. For Category 1 transfers, the message size will be defined before any transfers take place to allow the resources (memory etc.) to be allocated. In this way, the message size of a Category 1 transfer is unlimited, provided the required resources can be allocated. These resources must be allocated in all Users of the particular Category 1 transfer.

Since the Category 2 transfer does not define its resource requirements beforehand, the maximum size of the message may be limited, firstly by the resources allocated to be shared by the Category 2 transfers and secondly by the amount of these resources already being utilized by other Category 2 transfers. Since there is no cooperation between Users, different Users may not be able to allocate the necessary resources when the data transfer is received. For this reason, there is often a System Design message size parameter applied to the Category 2 transfers generally to limit the resources required by any individual transfer.

4.2.1.4 Packet Size

The real-time communications system shall provide support for a System Design parameter restricting the size of packets transmitted on the physical network. It is desired that the maximum size of a Category 2 message shall be independent of this parameter.

Rationale:

The maximum packet size is a system design parameter and so applies to both types of transfers. While the User may define the size of a message for either type of transfer to be as big as is required, other constraints may be applied to the way in which this message is transferred through the real-time communication system. One example of this is the data bus packet size, which defines the maximum size of an information unit for transfer across the data bus. This system design parameter may be required to limit the maximum size to less than that of the data bus standard to ensure latency requirements are maintained.

Messages larger than this size of either transfer type will be segmented into a number of data bus packets and reassembled into the message at the receiving node.

The real-time communication system should be capable of segmenting and reassembling Category 2 data transfers; otherwise the reduction of the maximum packet size may require a complete system redesign because of the reduced maximum size of these transfers. Thus, the two parameters need to be independent.

4.2.1.5 Flow Control

The real-time communication system shall allow the User to request flow control for a Category 1 data transfer.

Rationale:

Flow control of Category 1 data transfers allows the receiving User to prevent the data-providing User from sending further data until such a time when it may be accommodated. Flow control cannot exist for Category 2 data transfers because the receiving User does not know that further data is to be sent. For this transfer category it is acceptable to ignore data received but not for Category 1 transfers.

Flow control may also be required by layers within the real-time communication system for similar reasons. However, the layers will also have restrictions placed upon them by other data transfer parameters (e.g. latency) such that the maximum extent to which flow control may be used is limited. Thus sufficient resources will have to be supplied to support the transfer.

The flow control between Users is concerned with resources outside the real time communications system. Thus, the real-time communications system parameters do not apply to the use of flow control and the delay of information could be infinite; it is up to the system designer to define what is allowable. The real-time communications system is just providing a service to the receiving User to exercise flow control without the real-time communications system requiring extra resources to accept data from the data-providing User without passing it on to the receiving User simply because that User is not prepared to accept it.

Resources are allocated within the RTCP to transfer the data placed in the interface by User A according to the parameters defined when the Category 1 transfer was activated. See Figure 6. User B must accept data from the RTCP as was also part of the agreement. However, User B may not be able to handle the instantaneous data rate supplied by User A. For example, if the rate defined was 1000 blocks per second, it may be that the 1000 blocks occur in 1ms followed by 999ms silence. If User B can only handle 10 blocks at a time then, having received 8 blocks (assume 2 more in transit) in a short period of time, User B uses the flow control to prevent further transfers until some of the buffer space has been freed.

4.2.1.6 Flow Restriction

It is desirable that the real-time communications system shall allow the User to define a parameter for a Category 1 transfer defining the bandwidth of the series of messages to be passed. The User shall be restricted to this message flow bandwidth in subsequent transfers. The real-time communications system shall be capable of enforcing this flow restriction.

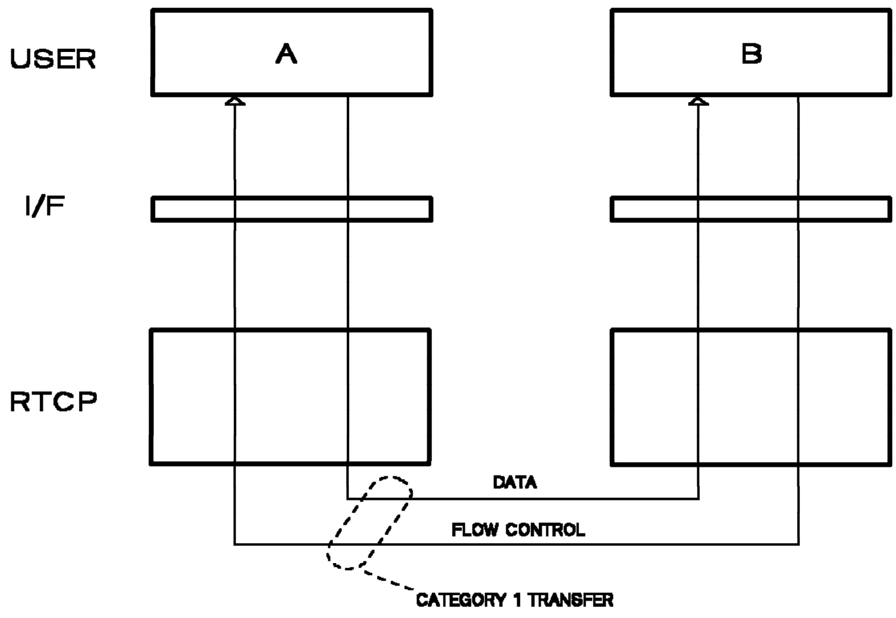


Figure 6. Flow Control

Upon acceptance of the message flow bandwidth, the real-time communications system shall reserve sufficient network bandwidth to guarantee the transfer of this message flow. It is also desirable that a system design parameter be defined restricting the maximum message bandwidth of all Category 2 transfers.

If these two desires cannot be implemented, it is required that the real-time communications system report to the User when excessive bandwidth requests result in the inability to meet all the data transfer parameters.

Rationale:

Flow control helps a receiving User manage its own resources, whilst flow restriction prevents a data-providing User from hogging more than its fair share of a global resource (RTCP).

While the Category 1 data transfer may be considered as a pipe from one User to another, the use of a multiplex network means that some resources (in this case network bandwidth) must be shared and will not be allocated solely to one data transfer.

The use of flow restriction is mainly applicable to Category 1 data transfers and is used to define the communication bandwidth resource which is allocated to the *particular transfer. While it is normally only considered as limiting the rate at which data may be transferred, it should also guarantee that the requested bandwidth is always available. Thus, each Category 1 transfer must compete with other transfers for the available bandwidth resource. Flow restriction is especially relevant for the lower layers of the real-time communications system in the case of a transfer where the message size is greater than the network packet size for the system. This will result in multiple packets being queued for transmission which could completely swamp the network. The flow restriction would allow the packets to only be presented for transfer at a particular rate, thus allowing other transfers to take place.*

The use of flow restriction may also be applied to Category 2 transfers, but only globally to all transfers of this category, not to individual transfers. The amount of bandwidth allocated to the Category 2 transfers would be allocated and the individual transfers would vie to use it.

The main problem of defining and implementing flow restriction is in the measurement of data bandwidth used/available; this is especially true for the Category 2 data transfers, the network traffic of which must be monitored.

From the above it is clear that flow restriction may not be feasible but is desirable.

4.2.1.7 Acknowledged and Non-Acknowledged Services

It is required that the RTCP provide support for both acknowledged and non-acknowledged Category 1 and Category 2 communications. This acknowledgement denotes that the information has been copied into the receiving User′s buffer (See Ack2, Figure 7).

Rationale:

Information transfer acknowledgements are only guaranteed if between processes. This means that acknowledgement is initiated when the information is presented to the destination process, not just when it enters the destination processor node. Process-to-process acknowledgement is at the User level and does not normally impact the RTCP. However, process-to-process acknowledgement may be implemented using RTCP functions as follows.

Within the RTCP, acknowledgement services will be required (e.g. to confirm the transfer of information for a Category 1 transfer.) These acknowledgements will be local to the RTCP and not visible to the data-providing User (Ack1), see Figure 7.

Use may be made of this acknowledgement (Ack1) to provide a notification to the data-providing User that the information has been transferred to the required User/RTCP interface, i.e., the point at which the RTCP has completed the information transfer (Ack2).

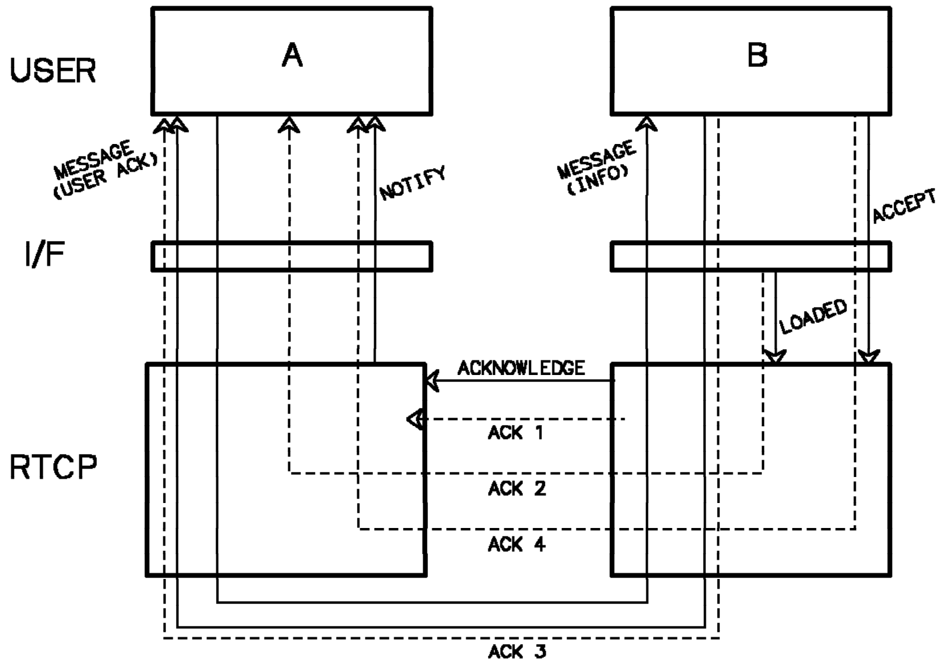


Figure 7. Types of Acknowledgements

The notification (Ack2) does not indicate that the receiving User has accepted or used the information (Ack3). This type of acknowledgement is normally considered to be provided by the data-providing User to indicate that the information is being used. This message will not be recognized as an acknowledgement by the RTCP.

However, this acknowledgement (Ack3) may also be provided using the RTCP acknowledgement (Ack1), shown as Ack4. To do this, the receiving User must indicate its acceptance of the information to its local RTCP entity, this can be acknowledged (Ack1) to the data-providing User's local RTCP entity, which will finally notify the data-providing User.

4.2.2 SYNCHRONIZATION SERVICE

It is required that the RTCP provide a mechanism that allows the synchronization of tasks processed by the application layer distributed at different remote User entity physical locations.

This system synchronization service can be accomplished by means of remote interrupts (4.2.2.1, "Remote Interrupts") or by means of a common knowledge of time (4.2.2.2, "Time Distribution").

4.2.2.1 Remote Interrupts

It is required that the RTCP provides a remote interrupt service that ensures the transfer of an interrupt vector to a destination entity or to a group of destination entities. It is required that the invocation of this service can be allocated to the RTCP (automatic remote interrupt.)

This does not imply a specific mechanism to implement the service.

Rationale:

The following provides examples of how interrupts might be used and shows, for instance, that interrupts at the RTCP level might be used by the User:

Remote interrupts to a destination User entity can be implemented via an interrupt request primitive from a User that is the source of the interrupt (Int1), see Figure 8.

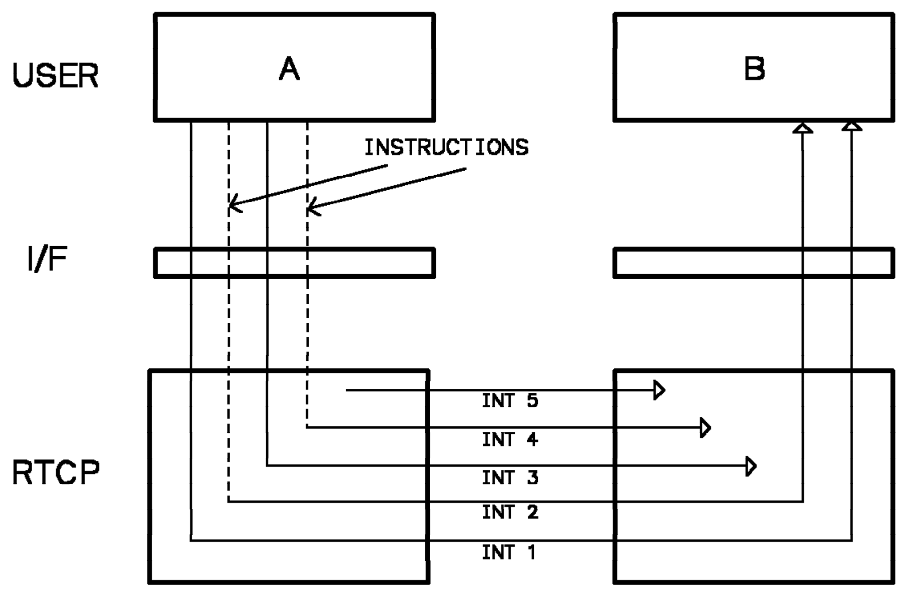


Figure 8. Types of Remote Interrupts

In addition, a User can instruct the RTCP to provide this interrupt to a remote User entity (Int2). A User can instruct the RTCP to provide this remote interrupt service *as a result of certain events in the network or periodically without involvement of the requesting User at the time, automatic invocation.*

A User can also request an interrupt to a remote RTCP entity (Int3), or instruct the service-providing RTCP to provide this via automatic invocation (Int4).

In addition, remote interrupts can originate in the RTCP to a remote RTCP entity (Int5).

4.2.2.2 Time Distribution

It is required that the RTCP provide a mechanism to initialize and maintain global time to be made available at the distributed RTCP/User interface.

Rationale:

The distribution of time within a system may be useful for process synchronization, event ordering, etc. However, this does not imply that network time is required for these functions. These functions could be implemented in a number of ways, including special network codes, interrupts, and periodic RTCP messages. The actual method used will depend on the accuracy and stability of time reference required (if at all).

4.2.2.3 Related Data Streams

The RTCP shall provide support to synchronize related data streams.

Rationale:

Certain related information may be transferred on separate data paths, such as voice and video. If so, synchrony of the sound with the image may not be given. Hence the RTCP must provide methods for reconstituting the timely delivery of related subsets of information.

4.2.3 MANAGEMENT SERVICES

It is required that the RTCP shall support a distributed control mechanism which invokes static and dynamic planned configurations. The RTCP shall also support a centralized control mechanism. The RTCP shall monitor status and keep fault logs for itself. It shall also support the handling of abnormal system states and exceptions. The RTCP shall support uninterrupted operational performance.

Rationale:

A distributed control management mechanism is necessary to support the needs and requirements of a distributed communications system, so that the loss of a single control entity does not totally disable communications.

This does not preclude the distributed control from being designed to follow a communications control algorithm driven by one entity, which presumably has back-up control schemes.

4.2.3.1 Configuration

It is required that the RTCP support preplanned communication configurations invoked by scheduled and unscheduled data events.

It is desired that the continuing and current state of these communication configurations and events be stored at multiple control points. This allows a distributed storage of static and dynamic preplanned system configurations, which prevents the loss of any single control point from impacting data scheduling.

It is required that the RTCP provide for static configuration of preplanned events and also provide for exception conditions via dynamic configuration control.

Rationale:

The RTCP intended systems are designed with predefined configurations that support both scheduled and unscheduled events. The RTCP must react to these events by enabling different configurations to meet the varying needs of the system.

The RTCP must react to exception system states. For instance, certain exception states may invoke different configurations based on semantic importance. These configurations may invoke selective load shedding.

4.2.3.2 Network Monitoring

It is required that the RTCP monitor the total network status and store the configuration. The network monitoring function shall be capable of being done at a single point. Monitoring information stored shall include the history of connections and state changes, with event or time labels. This monitoring function shall keep a running status of network health.

Rationale:

To provide a means of communication analysis and fault isolation, the RTCP monitors the status of all the network resources. This total log of RTCP state data is necessary to support the management control function, for instance, reconfigurations. It also provides a means for system evaluation of "bottlenecks" in the communications scheme.

4.2.3.3 Network Control

It is required that the RTCP provide a means to modify the operating modes of the entities comprising the network.

Rationale:

This provides dynamic control of the communications network for system requirements which may be changing in response to stimulus unknown to the communications network.

This control may drastically alter the operating modes of all the data links.

4.2.3.4 Intrinsic Maintenance

It is required that the RTCP utilize fault tolerant techniques to reschedule RTCP tasks to non-failed resources, thereby providing graceful network degradation.

The RTCP application shall utilize the fault log of static and dynamic communication faults stored and reported by the network monitoring function to detect and isolate the problems.

Faults introduced by improperly functioning portions of the network shall not be propagated beyond the RTCP. It is required that the RTCP supply the necessary detection to support this fault containment.

Rationale:

The network should, to the degree required by the system, enable alternate RTCP paths and resources. This fault tolerance scheme must have preplanned priorities to ensure that system performance maintains the desired level of integrity.

Improper communications or misdirected data and control, caused by a fault in the network, must be contained within the network to prevent higher level system failures.

4.2.3.5 Invoked Maintenance

It is required that the RTCP support an alternative preplanned communications path restructuring, which will allow a User to request an alternative path, if there has been a failure in a communications path.

Rationale:

This requirement provides for the possibility of path restructuring in the event that a communication path failure, which has not been detected by the RTCP, is observed by the user.

4.2.3.6 RTCP Reset

It is required that the RTCP support a reset function for the entire network and RTCP functions. This shall permit the RTCP to accomplish subsequent initialization and re-initialization.

It is required that following this function immediate rescheduling be allowed to accommodate different configurations.

Rationale:

This function allows the removal of communications lockups and "deadly embraces" of multiple scheduling functions requesting the same resource.

All scheduling functions and communications paths shall be reset/unlinked. Following verification of network status, scheduled RTCP tasks may be re-enabled.

5.0 USER/RTCP INTERFACE

This section describes the method by which the User/RTCP interface shall be defined.

Following paragraphs contain a general description of the interface between the User and RTCP. For a detailed description of this interface please refer to ″Logical Interface Definition for Real-Time Communication Protocols (RTCP)″, SAE ARD50033.

5.1 SERVICE PROCEDURE

The interaction at the layer interface between the User and the RTCP is characterized and may be defined by using service primitives, time sequence diagrams, and service parameters. Primitives provide the signalling mechanism for the interaction between User and the RTCP. Different types of primitives are defined depending on the character of the specific interaction. Time sequence diagrams define the discipline and procedure for using the primitives in the interaction. Service parameters further define implementation details as to the specific interaction.

5.2 SERVICE PRIMITIVES

Service primitives represent the interaction between the User(s) and the RTCP. Service primitives are conceptual in nature and, in particular, are independent of any kind of interface control implementation.

A primitive is presented in the form of an event, which is logically instantaneous and indivisible. It indicates the propagation direction, which may be either from a User to the RTCP or from an instance of the RTCP to a User. One or more parameters may be associated with a service primitive.

For the RTCP, five types of service primitives have been identified. The first four types described below have already been defined for the standard protocols associated with the OSI Reference Model. The fifth is specific to the real-time applications targeted by the RTCP.

5.2.1 REQUEST

The REQUEST primitive is issued by the User in order to request a particular service from the RTCP.

5.2.2 INDICATION

The INDICATION primitive is issued by the RTCP to indicate either that a service has been requested by a User or to indicate that an event has occurred in the network.

5.2.3 RESPONSE

The RESPONSE primitive is issued by the User in response to an INDICATION primitive.

5.2.4 CONFIRMATION

The CONFIRMATION primitive is issued by the RTCP to confirm the completion of a service.

5.2.5 INSTRUCTION

The INSTRUCTION primitive is issued by the User to specify to the RTCP what to do in case of particular events.

5.3 TIME SEQUENCE DIAGRAMS

Time sequence diagrams are used to illustrate how sequences of interactions are related in time. A time sequence diagram is partitioned by vertical lines into several parts, each area representing either a User or the RTCP. Arrows placed in the areas representing the service Users indicate the propagation direction of the specific primitive, i.e., to or from the User. The service providing (RTCP) area does not give any indication of the internal mechanisms, which are invisible to all Users.

Time sequence diagrams indicate the allowed sequence of events at each interface, and, where appropriate, the allowed sequence of events at all distributed User/RTCP interfaces which are solicited to provide the requested service.

At each User/RTCP interface the sequence of events is positioned along lines representing THE ELAPSE OF TIME EVOLVING DOWNWARDS. A cause and effect relationship between primitives at the same User/RTCP interface is symbolized by a dotted loop.

A cause and effect relationship between primitives at different User/RTCP interfaces is symbolized by a dotted line . This line does not presume in any way the protocol implemented in the RTCP layer.

An example of a time sequence diagram is shown in Figure 9. It may be noted that the vertical lines representing interfaces between Users and RTCP denote logical interfaces located in different sets of equipments.

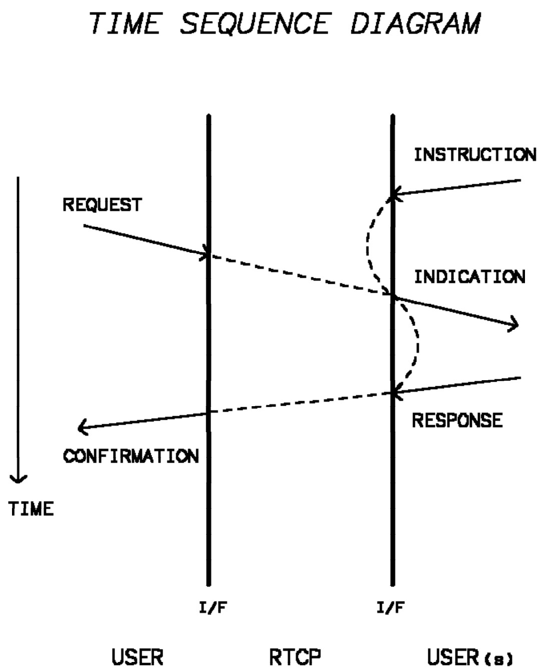


Figure 9. Sample Time Sequence Diagram

5.4 SERVICE PARAMETERS

Service parameters define the details for the implementation of the action associated with a specific service primitive. Depending upon the specific primitive, different parameters are used. Following subparagraphs describe some of the various parameters required. The list here of service parameters is not complete since the parameters depend on, and vary with, the specific interaction over the interface.

In addition to service parameters, a second category of primitives, interface control primitives, are required for a complete definition of the interaction between the User and the RTCP.

5.4.1 SERVICE TOPOLOGY

A number of the service parameters relate to the transfer topology, either explicitly, such as source address and destination address(es), or implicitly via a circuit ID that uniquely defines a previously established transfer topology. Other parameters may identify a specific message, for instance, Message ID.

5.4.2 QUALITY OF SERVICE

A large number of the service parameters can be grouped under the description service quality. They include such things as latency, whether one wants acknowledgement or not, security level, error control, flow control and restriction, priority, and delivery confidence.

5.4.3 RESULTS REPORTING

Another category of primitives are used to further define the result of a completed action; a report. These are used to notify the User of the success/failure of a completion of a request.

6.0 REFERENCES

1. Information Processing Systems - Open Systems Interconnection - Basic Reference Model.

   ISO 7498

2. Information Processing Systems - Open Systems Interconnection - Basic Reference Model.

   Addendum 1: Connectionless-mode Transmission

   ISO 7498 Addendum 1

3. High Speed Linear Bus Standard, Issued August 29, 1988

   SAE AS4074.1 (Being revised as AS4074)

4. High Speed Ring Bus Standard, Issued August 29, 1988

   SAE AS4074.2 (Being revised as AS4075)

5. Logical Interface Definition for Real-Time Communication Protocols (RTCP), Under development.

   SAE ARD50033

Appendix A. TARGET APPLICATIONS KEY DESCRIPTOR WORDS

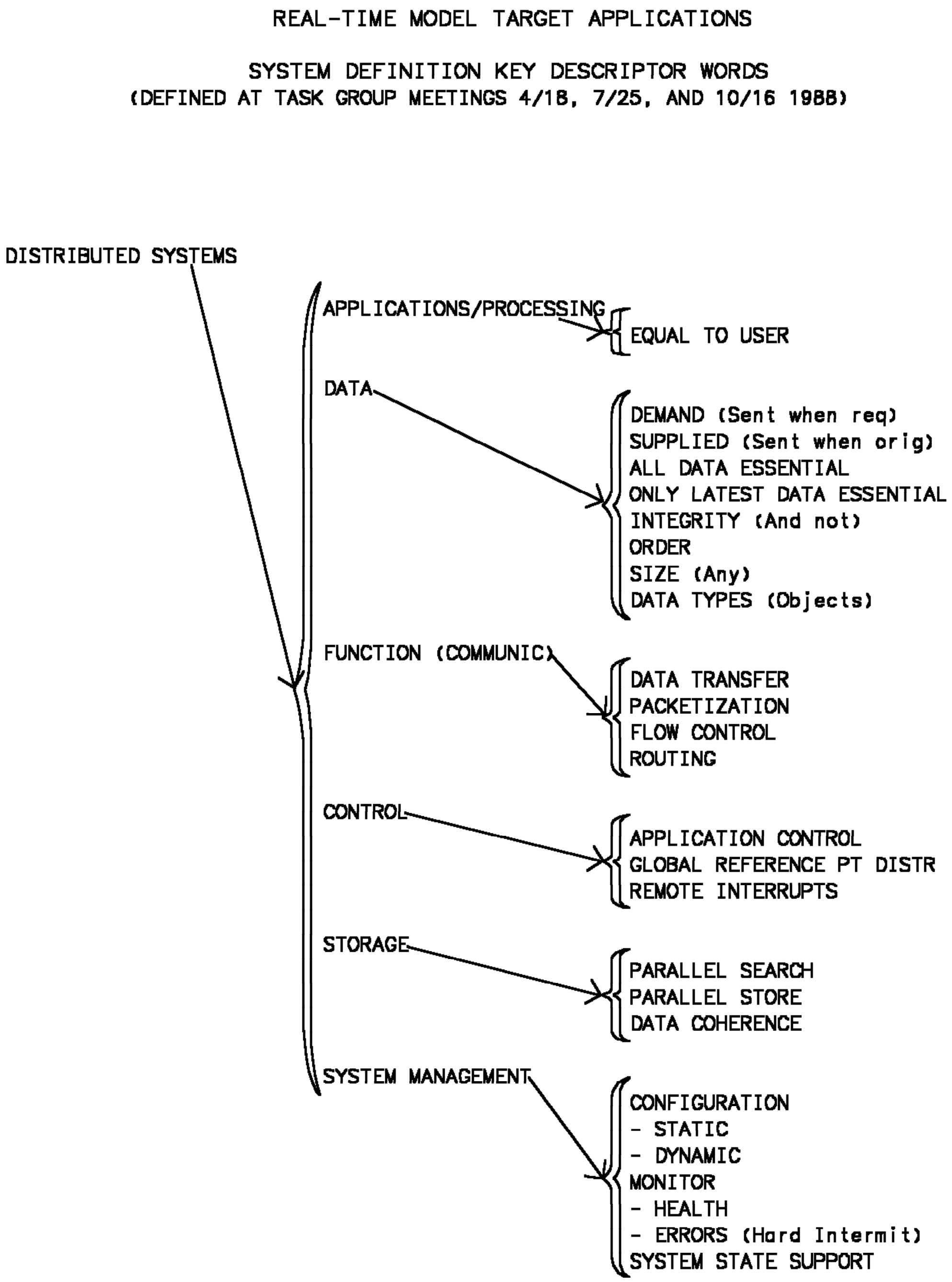


Figure 10. Key Descriptor Words

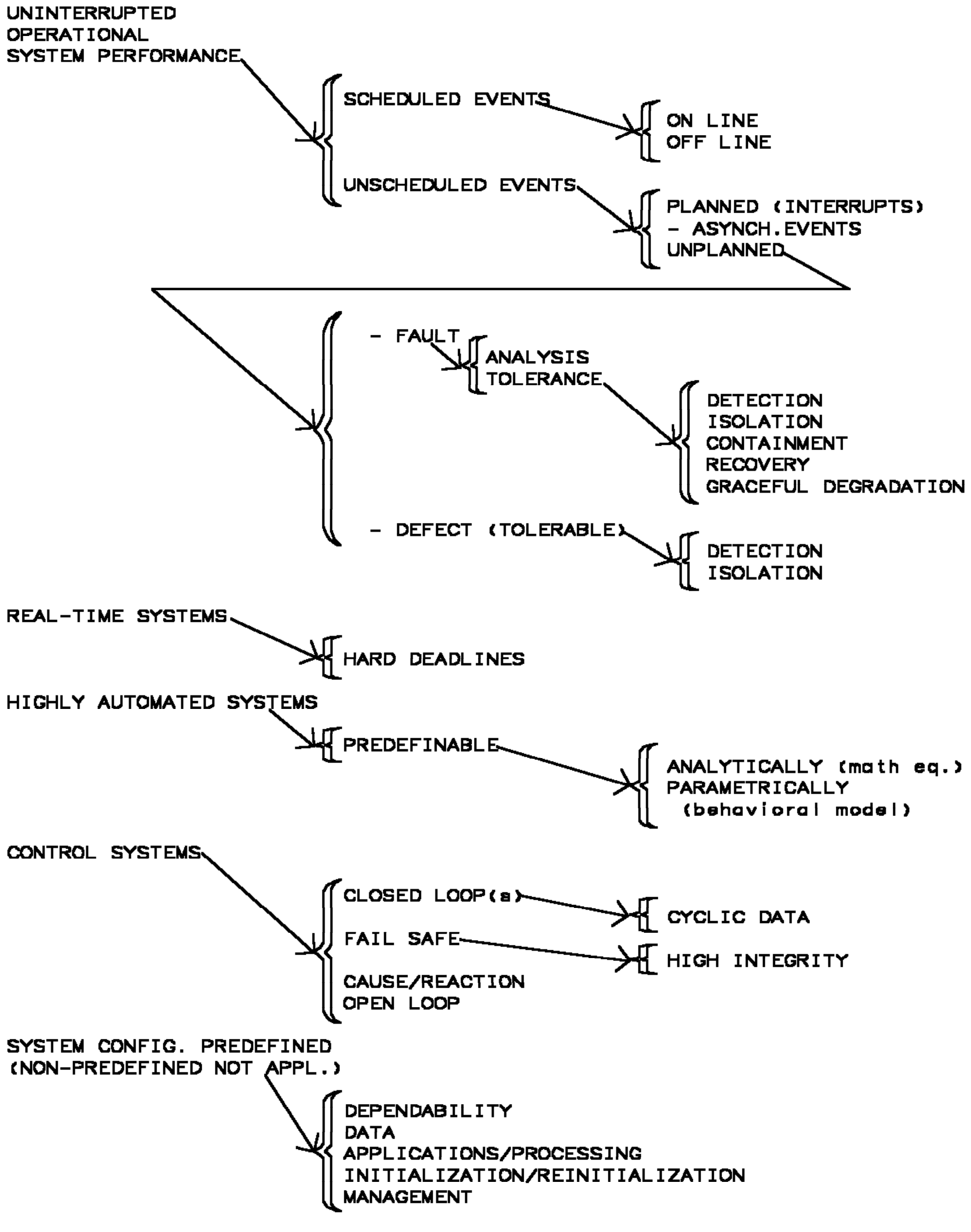


Figure 11. Key Descriptor Words cont'd.