CCTV to Access Control Standard - Message Set for System Integration

SIA TVAC-01-2001.04

Sponsor
Security Industry Association

Copyright © 1999-2001 SIA

Publication Order Number: 14052

FOREWORD

This standards document is published by the Security Industry Association (SIA) and was developed and adopted by a consensus of industry volunteers in accordance with SIA's standards development policies and procedures. It is intended to facilitate product compatibility and interchangeability, to reduce misunderstandings between manufacturers and purchasers, and to assist purchasers in obtaining the proper products to fulfill their particular needs.

The existence of this or any SIA standards document shall not prevent any SIA member or non-member from manufacturing, selling, or using products not conforming to this or any SIA standard. SIA standards are voluntary. SIA encourages the use of this document but will not take any action to ensure compliance with this or any other SIA Standard.

SIA assumes no responsibility for the use, application or misapplication of this document. Industry members using this document, particularly those having participated in its development and adoption, are considered by SIA to have waived any right they might otherwise have had to assert claims against SIA regarding the development process of this standard.

Although some SIA standards establish minimum performance requirements, they are intended neither to preclude additional product features or functions nor to act as a maximum performance limit. Any product the specifications of which meet the minimum requirements of a SIA standard shall be considered in compliance with that standard. Any product the specifications of which exceed the minimum requirements of a SIA standard shall also be considered in compliance with the standard, provided that such product specifications do not exceed any maximum requirements set by the standard. SIA standards are not intended to supersede any recommended procedures set by a manufacturer for its products.

SIA reserves the right to revise this document at any time. Because SIA policy requires that every standard be reviewed periodically and be either revised, reaffirmed, or withdrawn, users of this document are cautioned to obtain and use the most recent edition of this standard. Current information regarding the revision level or status of this or any other SIA standard may be obtained by contacting SIA.

Requests to modify this document are welcome at any time from any party, regardless of membership affiliation with SIA. Such requests, which must be in writing and sent to the address set forth below, must clearly identify the document and text subject to the proposed modification and should include a draft of proposed changes with supporting comments. Such requests will be considered in accordance with SIA's standards development policies and procedures.

Written requests for interpretations of a SIA standard will be considered in accordance with SIA's standards development policies and procedures. While it is the practice of SIA staff to process an interpretation request quickly, immediate responses may not be possible since it is often necessary for the appropriate standards subcommittee to review the request and develop an appropriate interpretation.

Requests to modify a standard, requests for interpretations of a standard, or any other comments are welcome and may be sent to:

Standards
Security Industry Association
635 Slaters Lane, Suite 110
Alexandria, VA, 22314

E-mail:
Standards@SIAOnline.org

This document is owned by the Security Industry Association and may not be reproduced, in whole or part, without the prior written permission from SIA.

ACKNOWLEDGMENTS

This standard was developed by the SIA TVAC Standards Subcommittee. SIA gratefully acknowledges the efforts of the security industry volunteers on the Subcommittee developing this standard.

Chairman of the SIA Standards Committee:
ADT Security Services, Inc. ..... Bill Moody

Chairman of the SIA TVAC Standards Subcommittee:
Integrated Command Software, Inc. ..... R. Hunter Knight

Contributors to the SIA TVAC Standards Subcommittee:
Pelco ..... Stephen Robinson
Salient Systems..... Tim Wegesin
Salient Systems..... Per Hanssen

SIA Liaison..... Wadei Powell
Mark A. Visbal

This standard was approved by open industry vote (88% majority) on May 4, 2001

Advantor Corp..... Justin Cullen
Digital Vision Systems..... Mark J. Young
Integrated Command Software ..... Hunter Knight
Pelco..... Dave Smith
Salient Systems Corp..... Per Hanssen
Teklink Security..... Fred Mu
Ultrak..... Raymond Parfett
Vicon Industries..... Raymond Lyons

Revision History

The following are changes made to this document, listed by revision.

APRIL 2001 BASELINE

Original Publication

Table Of Contents

CCTV to Access Control Standard - Message Set for System Integration
SIA TVAC-01

1. SCOPE

This standard details an interface protocol, including a general message architecture and specific message sets, for communications between a CCTV switching sub-system and an Access Control sub-system . This standard may be applied to any two or more sub-systems of either type. This standard may also be applied to sub-systems of other types if their functionality is supported within the message sets.

This standard assumes bi-directional communications where no sub-system is a master over the other/s. Each sub-system may issue any of the defined messages, however each sub-system may decline requests that are made from the other.

This standard specifies the high order layers of the interface protocol including: Application, Presentation, and Session. It is assumed that other conventional protocols, such as TCP/IP, RS-232, and RS-485, will be used for the lower protocol layers, including: Transport, Network, Data Link, and Physical.

The message architecture in this standard is flexible and intended to be changed over time, however it is intended that additional messages will be added to the message set by formal revision of this document.

This standard is intended for use by manufacturers in the design of CCTV, Access Control, and other similar systems in the security industry and related industries. It is also intended for reference by all affected parties, including security system installers and programmers, system specifiers, security managers, and other end users.

This standard is voluntary.

2. REFERENCE DOCUMENTS

2.1 RELATED AREAS

Additional guidance on areas relating to this standard, as noted and otherwise, can be obtained from the sources listed below.

- *NTCIP 1205Recommended Standard – Object Definitions for Closed Circuit Television (CCTV)*
- *Principles of Database Management by James Martin (Prentice Hall, 1976)*

2.2 PRECEDENCE

In the event of conflict between this standard and other reference documents mentioned herein, the order of precedence shall be:

1) this standard
2) other reference documents

3. CONVENTIONS AND DEFINITIONS

3.1 CONVENTIONS

3.1.1 Units of Measurement.

In accordance with SIA Policy, the units of measurements used throughout this publication are the units of the System International d' Unites (SI), commonly known as metric units. Equivalent English Units, enclosed in parenthesis, are also

used in this publication. These equivalent English Units are approximate conversions and are provided for easy reference.

3.1.2 Tolerances.

Unless otherwise specified, the tolerance for measurements specified within this standard shall be 10 percent (±10%).

3.1.3 Nomenclature and Identification of Sections

Sections within this standard are identified and referenced by the number preceding each section. Unless otherwise specified, references to a section refer to only that section and not to subsequent subsections within the section.

3.1.4 Binding Language

This standard uses the term "shall", "will", and "must" to convey binding requirements.

The term "may" is used to convey features that are allowed but not required.

Terms such as "is", "are", "will", and others are used to convey statements of fact for advisory purposes only.

The annotation "Note:" also precedes advisory information.

Where this standard is silent on a feature, the feature is permitted so long as it is not in conflict with the requirements contained herein.

Appendices contain non-binding information.

3.2 DEFINITIONS

For the purpose of this standard, the following terms and acronyms have the meaning indicated.

data item – A specific component of a record in a database that specifies the value of an attribute of a device or component represented by a record in a database.

key – An identifier of a record in a table. The key may be unique, specifying a single record in the table, or non-unique, specifying a 0, 1 or some records in the table. A key is composed of one or more data items in the table taken in a specific order.

message – A message is a collection of characters that meet the format as defined in this specification.

NLS - National Language System

pointer – a data value that in whole or in part identifies a specific record in a database

record – A group of related data items that, taken together, describe a device or component.

schema - The overall definition of the relationships between the data items, record definitions, occurrences, and content including the rules for validation of the items that make up the database. Specification Overview

system – A collection of devices and software that utilizes this specification to communicate with other entities.

4. SPECIFICATION OVERVIEW

These specifications detail the application, presentation, and session layers of the interface protocol.

The message traffic allows the following functionality between the sub-systems:

(a) Establish a connection

(b) Synchronize databases (reconcile capabilities or negotiate control)

(c) Issue a command for action

(d) Advise of actions, events, or other activity

4.1 INTERFACE LEVELS

This specification defines a multi-level interface definition where each successive level includes the

capabilities of lower levels. Within the specification, each capability (represented by a message type or message data item/s) is categorized to indicate the level of interface where that capability must be supported, as shown in the message content tables. (See Section 7)

This specification establishes the required message types for the different levels of interface implementation. An interface must fully support all of the specified data items and message types of a level to qualify as supporting a particular level of the standard. Some message types include data items from multiple levels of interface capabilities

At a lower level, the functionality associated with a data item from a higher level need not be supported, however the data element must be present in the supported message. The value of a data item of a higher level of interface, not supported in a lower level of the interface, shall be spaces. Thus message structures do not vary with the level of the interface. This approach permits exchange between disparate systems of different levels.

The following list indicates the levels.

- **B – Basic**: A part of the simplest, least functional level of the interface. Support for 'S' data items in 'B' tables need not exist for 'B' certification however, messages with 'S' fields in 'B' records must reserve space for 'S' data items.
- **S – Standard**: A part of standard level of interface providing a reasonable level of interface capability. S includes B items so any B item is also a requirement for an S level interface.
- **F – Future**: This level of implementation is reserved for future applications. Some of these areas are identified throughout this document. As this standard is utilized, additional areas of functionality will become evident. It is the intention of this subcommittee that this document remain a "living" document.

In order to ensure compatibility of sub-systems at these discrete levels, there are no mixed levels of compliance. A system that meets all B requirements, but not all S requirements, is a B level capability system.

Manufacturer extensions are not provided for in this standard. Any desired additions should be incorporated into the standard by formal revision.

4.2 ASSUMPTIONS

The following assumptions have been made in the architecture of this specification:

(a) All devices and sub-system ID's are unique

(b) Each sub-system is responsible for defining its own events and providing a legend to the subsystem it is addressing

(c) No limit on message size is imposed by this standard, but there may be practical limitations outside of this standard.

(d) A sub-system does not have to comply with a command or request, but it must respond to the command or request.

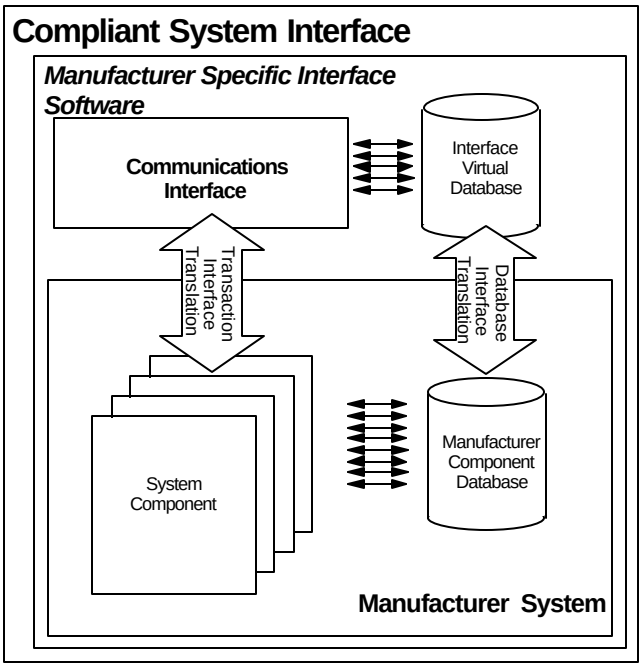
(e) Messages are provided to support system initialization and the message sequence for connection initialization and log in.

(f) All messages must be acknowledged within the time out specified for the specific connection.

4.3 TYPICAL SYSTEM LAYOUT

Figure 1 shows a typical layout for implementation of this protocol within a sub-system . A manufacturer's system is encapsulated by a layer of software that implements this interface protocol. The virtual database of the interface is supported by software as is the command, event reporting, and communications interface.

Figure 1: System Implementation



4.4 CHARACTER SET

All message characters will be ASCII displayable and printable except as specified in this specification. Special characters without a well-known ASCII representation and keyboard entry will not be a part of the message character set.

The allowed message character set is provided in Table 1. Fundamentally, the character set is limited to 7-bit ASCII in an 8-bit byte. Titles may be Unicode.

This area intentionally left blank

Table 1: Allowed Characters

ASCII	HEX	Description
A	41	UPPER CASE A
B	42	UPPER CASE B
C	43	UPPER CASE C
D	44	UPPER CASE D
E	45	UPPER CASE E
F	46	UPPER CASE F
G	47	UPPER CASE G
H	48	UPPER CASE H
I	49	UPPER CASE I
J	4A	UPPER CASE J
K	4B	UPPER CASE K
L	4C	UPPER CASE L
M	4D	UPPER CASE M
N	4E	UPPER CASE N
O	4F	UPPER CASE O
P	50	UPPER CASE P
Q	51	UPPER CASE Q
R	52	UPPER CASE R
S	53	UPPER CASE S
T	54	UPPER CASE T
U	55	UPPER CASE U
V	56	UPPER CASE V
W	57	UPPER CASE W
X	58	UPPER CASE X
Y	59	UPPER CASE Y
Z	5A	UPPER CASE Z
a	61	Lower case a
b	62	Lower case b
c	63	Lower case c
d	64	Lower case d

ASCII	HEX	Description
e	65	Lower case e
f	66	Lower case f
g	67	Lower case g
h	68	Lower case h
i	69	Lower case i
j	6A	Lower case j
k	6B	Lower case k
l	6C	Lower case l
m	6D	Lower case m
n	6E	Lower case n
o	6F	Lower case o
p	70	Lower case p
q	71	Lower case q
r	72	Lower case r
s	73	Lower case s
t	74	Lower case t
u	75	Lower case u
v	76	Lower case v
w	77	Lower case w
x	78	Lower case x
y	79	Lower case y
z	7A	Lower case z
	20	Space (SP)
0	30	Zero
1	31	One
2	32	Two
3	33	Three
4	34	Four
5	35	Five
6	36	Six

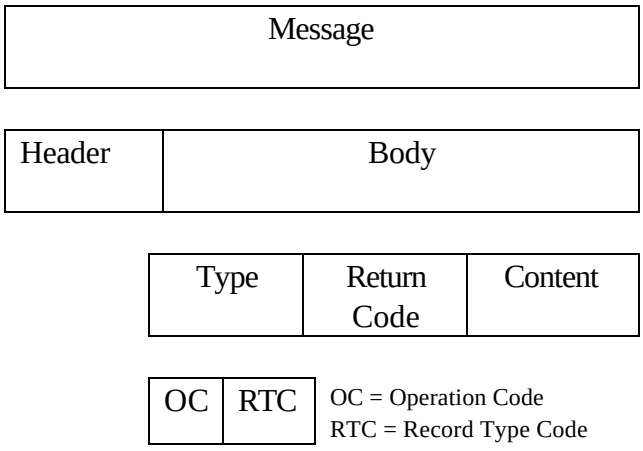
ASCII	HEX	Description
7	37	Seven
8	38	Eight
9	39	Nine
!	21	Exclamation Point
@	40	“at” symbol
#	23	Number sign
$	24	Dollar sign
%	25	Percent
^	5E	Caret
&	26	Ampersand
*	2A	Asterisk
(	28	Left parenthesis
)	29	Right parenthesis
+	2B	Plus sign
-	2D	Minus sign
=	3D	Equal sign
\	5C	Left or Back slash
/	2F	Right or Front slash
?	3F	Question mark
>	3E	Greater than
<	3C	Less than
,	2C	Comma
.	2E	Period
[	5B	Left bracket
]	5D	Right bracket
{	7B	Left brace
}	7D	Right brace

5. MESSAGE STRUCTURE

Each message has a common message header structure and an individual message body. The purpose of the header is to support applications level routing, identification, flow-control, and error detection. The purpose of the body is to transmit information.

Figure 2 shows the relationship between the message components in block diagram form. Tables 2 - 6 that follow provide specific details of the structures defined in the message block diagram.

Figure 2: Message Structure Block Diagram



This area left intentionally blank

5.1 MESSAGE HEADER

The message header contains the data items defined in the Table 2.

A message header must contain all data items and only those data items shown for a given level of implementation. For example, a basic B-Level implementation must contain a **message_start** and **checksum**, but no other data items.

The data item marked (reserved) should be filled with spaces as place holders.

Table 2: Message Header (B)

Item	Level	Data Item	LEN	Description
1	B	message_start	1	A distinctive beginning of message character. STX or 0x02 shall be the start character. This is the only typically non-keyboard character to be supported.
2	B	message_length	4	The length in characters of the message including the header and message body. Range is 0037 to 9999.
3	B	version	2	Version of the protocol used in this message.
4	B	destination_id	10	The identifying name of the destination to which the message is being sent.
5	B	source_id	10	The identifying name of the source from which the message is being sent.
6	B	(reserved)	6	Reserved for the future
7	B	checksum	4	An ASCII version of the checksum for the entire message including the message body but not including the checksum. The checksum will be CRC16 where the checksum will be represented in four ASCII bytes. If the checksum is 0xC7D5, the four bytes will be the characters : 'C7D5' In calculating the checksum, the order is MSB first in the application. No assumption should be inferred about bit order in the transmission. See appendix 'A' for CRC example. The protocol assumes that any message which fails the checksum test contains no valid content.

5.2 MESSAGE BODY

The following pages define the message body for the envisioned set of messages. These messages maintain a common basic format: **message_type**, **message_return_code**, and **message_content**, as shown in Table 3 below.

Table 3: Message Body (B)

Field	Length	Remark
message_type	4	See the operation codes and record type codes in the two lists below. Message types are constructed from these two codes.
message_return_code	6	A six-character reservation of space for the exchange of status information.
message_content	Variable	Varies with record type codes.

Message_type is a four-character message identifier built from a one character **operation_code** and a three character **record_type code**. It specifies the type of message. This document defines each message type that is to be a part of the specification. Many of these messages relate to the definition of objects such as views, monitors, titles, etc. A switching system will normally have many of these objects in its databases.

Message_return_code is a six byte field used to define the number or order of records in different types of messages or to convey the result code or status of a transaction. A blank message_return_code means status is not relevant.

Message_content is a structure defined by the record_type code. The length varies depending on record type.

5.2.1 Operation Codes

The **operation_code** defines the type of transaction. The allowed operation codes are shown in the table below. Not all operation codes are applicable to all record types. Valid combinations of operation_code and record_type_code are defined later under record types.

Table 4: Operation Codes (B)

Operation Code	Operation Name	Operation Effect
S	Status Request	The status request operation code is used to communicate information and not to manage a database. It is used to establish and manage connections, to issue command requests, and to report activity. The specific functionality required of a SR message is determined by the record type code portion of the message body header. The response operation code to SR messages is a MA response message. See Session and Message Exchange and Management for the details of this process.
Q	Query Request	This operation code is used to request records of the specified type that have field values matching any field values specified in the request message. If no non-blank fields are provided it means all records will be returned. See Session and Message Exchange and Management for the details of this process.
U	Update Request	This operation code is used to insert or update a component database record on a

Operation Code	Operation Name	Operation Effect
		destination sub-system . If the record transmitted has a unique identifier that does not match any record in the destination data base, the record will be inserted. Otherwise it will be updated. Update requests from sub-system A to sub-system B to modify a sub-system B record from a sub-system C will be denied. See Session and Message Exchange and Management for the details of this process.
D	Delete Request	This operation code is used to delete a component database record on a destination sub-system . If the record transmitted has a unique identifier that does not match any record in the destination data base, message will be rejected. Otherwise, the specified record will be deleted. Delete requests from sub-system A to sub-system B to delete a sub-system B record from a sub-system C will be denied. See Session and Message Exchange and Management for the details of this process.
A	Message Answer	This operation code is used to respond to messages. There is no data content included in this message. See Session and Message Exchange and Management for the details of this process.

5.2.2 Record Type Codes

Each record type data item is defined by a two letter code in the lists below. Message types are created by concatenation of an operation code with a record type code.

This table is ordered by record type code. The Section # refers to the location in Section 7 of this document where a detailed description is found for each record type. Section 7 is ordered in functional groupings of record types.

Table 5: Record Type Codes

Record Type Code	Record Type	Section #
AR	Activate Event Reporting Command	7.3.8
BA	Badge Access Command	7.3.7
CA	Cameras Table	7.2.4
CC	Close Connection Command	7.1.3
CL	Configure Logging Command	7.2.24
CS	Connection Status Table	7.1.1
EA	Event Acknowledge Command	7.3.11
EC	Event Code Table	7.2.15
EM	Event Report Message	7.4.1

Record Type Code	Record Type	Section #
ER	Event Response Table	7.2.17
HL	Holidays List Table	7.2.12
IP	Input Point Table	7.2.21
LI	Login / Logout Message	7.1.4
LM	Log Message	7.4.2
MC	Movement Command	7.3.2
MD	Motion Definition File Table	7.2.2
ML	Monitor Lists Table	7.2.8
MN	Monitors Table	7.2.7
OC	Output Control Command	7.3.5
OP	Output Point Table	7.2.22
OT	Operation Code Table	5.2.1
PC	Portal Control Commands	7.3.6
PL	Privilege Level Table	7.1.27
PO	Poll Message	*7.1.5*
PR	Preset Definition Table	7.2.3
RC	Receiver / Positioning System Table	7.2.19
RD	Response Definition Table	7.2.16
RV	Retrieve Video Record	7.3.10
SA	Salvos Table	7.2.9
SC	Salvo Control Command	7.3.4
SD	Schedule Definition Table	7.2.13
SE	Scheduled Event Table	7.2.14
SP	System Parameters Table	7.1.2
ST	Sector Title Table	7.2.20
SV	Save Video Record	7.3.9
TC	Tour Control Command	7.3.3
TD	Time and Date Command	7.3.13
TL	Titles Table	7.2.5
TP	Title Display Parameters Table	7.2.6
TR	Tours Table	7.2.10
TS	Tours Steps	7.2.11
UI	User Interface Table	7.2.18

Record Type Code	Record Type	Section #
VC	View Display Command	7.3.1
VW	Views Table	7.2.1
XR	Cancel Event Reporting Command	7.3.12

5.2.3 Message Return Codes

The following table provides the message return codes for the interface.

All messages received require a reply. These codes are used to indicate the status of a receiving sub-system or device relevant to a message it has received. The NA code is used to send an initiating message.

Table 6: Return Codes

Return Code	Meaning
NA	No Applicable Status. Field is left justified, right space filled.
OK	Last message received and processed OK. Field is left justified, right space filled.
NOxxxx	Last Message Received but not processed OK. xxxx is an optional reason code defined in table 6.1 Field is left justified, right space filled.
RJxxxx	Last Message Received but Rejected. xxxx is an optional reason code defined in table 6.1 Field is left justified, right space filled.
nnnnnn	If this is a number in character format, then it represents the number of records left to send including this record. 0 means no records found, 1 would indicate that this is the last record. When used as a counter, the format of this field is right justified and left zero filled.

Table 6.1: Reason Codes

Return Code	Meaning
okok	OK
ukcm	Unknown command
dnip	Denied due to insufficient privilege
rjrj	Rejected
ivrc	Invalid referential context: When a data base update is received that cannot be implemented due to a referential context problem.
ooss	Out of Storage Space

6. MESSAGE CONTENT - UNDERLYING DATABASE SCHEMA

A record type defines objects (things) like views, monitors, etc., may be thought of as relating to files with a structure matching the object which is being described. The relationship between the records of such related files defines many of the properties of this interface.

All of the messages fall into the four categories described below.

6.1 SESSION MANAGEMENT MESSAGE

These messages provide the tools for interface and connection management and control functionality. They do not convey information, events, commands, data updates, etc., associated with operational use of the interface.

6.2 COMPONENT DATABASE MESSAGE

These messages provide the functionality to manipulate the data that defines all of the system components.

6.3 COMMAND MESSAGE

These messages provide the functionality to direct the operation of a connected system. Conceptually, these are system instructions to perform specific tasks associated with using the interface but not including management of the session or maintaining data.

6.4 EVENT MESSAGE

These messages provide the functionality for reporting events to a connected system. These messages do not instruct the behavior of the systems although a system receiving an event message might initiate some action based on its programming. Again, session management messages and database management are excluded from this category.

7. MESSAGE CONTENT - TABLES AND CONVENTIONS

The tables that follow define the data structures of the component data items of the interface. They define the common types of items (objects) which must be defined to support the interface.
The table column headings are as follows:

Item: The item number of a data element within a particular objects data structure.

Level: The proposed level of interface where this data item must be supported. The allowed values are B – Basic, S- Standard, and F- Reserved for Future development.

If a data item is specified as a level B, it shall also be a part of level S. If a data item is specified as part of level S or B, it shall also be a part of level F.

Data Item: The data item names are descriptive, usually lowercase, words concatenated with the underscore (_) character. Abbreviations are avoided unless they are essential. The first two

characters define the table of which the record is a part.

Recalling that the interface protocol is designed around a database schema that will support a set of capabilities and exchange of commands, data item names use the following conventions:

a) If the data item name is shown in boldface print **(BOLD)**, it is a part of the key to a record in this table if a key exists. If it is followed by a number, the number specifies of which key the data item is a part and the order of that data item in the definition of the key in the case of multi-data-item keys. The format of the specification is [key number. order]. Key number may be 1, 2, etc. depending on the number of keys to the table. If the brackets are square [ ], then the key must be unique. If the brackets are curly { }, the key is not unique. Where there is only one field and one key, the key may be assumed not to be unique unless it is followed by empty square brackets []. Where a data item is a part of more than one key, then it will be followed by multiple brackets containing number pairs.

b) if the data item name is lower case, not followed by brackets, and not bold, it is a property of the object represented by the data structure.

c) If the data item name is all UPPER CASE, it is a pointer or part of a pointer into another table denoted by the record type in (parenthesis) following the field name. The data item may also be a key as prescribed by a), above.

d) All data items are in character format. There are no integer, decimal, etc. formats used.

LEN: Provides the length in characters of the data item. All data items are ASCII printable/displayable characters. All data items are fixed length fields, and are padded to the length of the field as detailed below in 'value' / Format conventions.

'value' / Format: This column either provides a list of allowed values enclosed in single quotes ('), the location of such a list, or a format specification for the values of the data item.

Specifications for 'value' / Format use the following conventions:

(a) **C** - denotes a character only field where any acceptable character may be used. The characters will be left justified in the field and right blank filled to the field length.

(b) **c** - denotes a character data item which must be a valid alphabetic or counting integer including the 0 character. Thus, for example, cccccc denotes a six character field containing alpha/numeric characters. Special characters are excluded except for blanks. c fields are left justified, right blank filled.

(c) **n** - denotes a character data item which must be a counting integer including 0. Thus, for example, nnnnnn denotes a numerical value expressed as six

characters. n fields are right justified, left zero filled.

(d) **b** - denotes a character data item which must be a character, 0 and/or 1. Thus, for example, bbbbbb denotes a string of six 0s and/or 1s. There is no justification rule for this type of field. All positions are taken to be significant.

(e) Any character other than **n**, **c**, or **o** in a 'value' / format specification shall be taken as a literal. For Example: nnn.nnn.nnn might look like 123.345.567 in the table. The embedded periods are a part of the data item.

Description: The description summarizes the meaning and interpretation of the effect of this data item.

7.1 SESSION MANAGEMENT MESSAGES

7.1.1 Connection Status Table – CS (B)

This message content includes the expected communications interface specification as a part of the message. It is used to establish a connection and to report the connection status with a particular subsystem. In the chapter, Message Dialog the exchange process to full implement a connection is explained. In each case the source address in the message header is the name of the message sender.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	cs_connection_type	2	'1','2', '3'	1 – Serial Port, ASCII, 8 bits, No Parity,1 Start Bit, No Stop Bits: 2 – Ethernet LAN, TCP 3 – Ethernet LAN, UDP:
2	B	cs_port	2	nn	If cs_connection_type =1, the port number of the connection to the referenced sub-system .
3	B	cs_speed	5	C	Allowed values for cs_speed are : • .3K • 1.2K • 2.4K • 4.8K • 9.6K • 19.2K • 38.4K • 56K • 64K • 128K • 256K
4	B	cs_host_name	101	C	If cs_connection_type =2 or 3, the host name of the remote sub-system of this record.
5	B	cs_service_name	51	C	If cs_connection_type =2 or 3, the service name of the remote sub-system of this record.

Item	Level	Data Item	LEN	'value' / Format	Description
6	B	cs_status	1	See List	Report the status of the connection: • U – Unknown • M – U Message Received & M Answer Sent • A – M Received & A Answer Sent • R – Ready States M, A, and R are related by the session startup procedure described in Section 8.
7	B	cs_error_condition	2	See List	Reports the details of the unknown (U) status of the connection. • OK – Connection OK • UK - Unknown • TO - Timed Out • CL - Cancelled Locally, Operator • RL - Rejected Locally • CR - Cancelled Remotely • RS – Remote Shutdown • NS – Normal Shutdown • RR - Rejected Remotely
8	B	cs_ sub-system _id	10	C	The identifying name of the subsystem for this connection record.
9	B	cs_poll_interval	4	nnnn	The poll interval of the specified sub-system .
10	B	cs_time_out_interval	6	nnnnnn	A time interval in 10 millisecond increments after which an anticipated response shall be considered to have not occurred.
11	B	cs_retries	2	nn	The minimum number of retries before declaring a connection down. After number of retries is exceeded the connection status (cs_status) will be set to "U" and the cs_error_condition will be set to TO.
11	B	cs_time_zone	2	nn	Local time zone from 0 to 23 from International Date Line
12	B	cs_dlst_value	3	nnn	Current minutes ahead or behind standard time for this time zone.
13	B	cs_dlst_direction	1	b	1 – Ahead, 0 – Behind, cs_dlst_value is ahead or behind local standard time.

7.1.2 System Parameters Table - SP (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	sp_level	1	'B','S'	Level of interface implemented
2	B	sp_version	4		This field contains the interface implementation version proposed by the sending system.
3	B	sp_model	32		Manufacturer's unique ID for this product
4	B	sp_CS	10		System Configuration Implementation Level
5	B	sp_PO	10		Poll Implementation Level
6	B	sp_CC	10		Closed Connection
7	B	sp_LI	10		Login Implementation Level
9	B	sp_CP	10		Change Password Implementation Level
10	B	sp_OM	10		Operating Mode Implementation Level
11	B	sp_TD	10		Time and Date Implementation Level
12	B	sp_PL	10		Privilege Level Implementation Level
13	B	sp_CL	10		Configure Logging Implementation Level
14	B	sp_VW	10		Views Implementation Level
15	B	sp_CA	10		Cameras Implementation Level
16	B	sp_TL	10		Titles Implementation Level
17	B	sp_MN	10		Monitors Implementation Level
18	B	sp_ML	10		Monitor Lists Implementation Level
19	B	sp_SA	10		Salvos Implementation Level
20	B	sp_TR	10		Tours Implementation Level
21	B	sp_SC	10		Schedules Implementation Level
22	B	sp_SE	10		Scheduled Event Implementation Level
23	B	sp_EV	10		Event Implementation Level
24	B	sp_EA	10		Event Action Implementation Level
25	B	sp_KP	10		Keypad Implementation Level
26	B	sp_KM	10		Keypad Monitor Implementation Level
27	B	sp_KV	10		Keypad View Implementation Level
28	B	sp_KR	10		Keypad Receiver Implementation Level
29	B	sp_PS	10		Receiver / Positioning System Implementation Level

Item	Level	Data Item	LEN	'value' / Format	Description
30	B	sp_ST	10		Sector Title Implementation Level
31	B	sp_IP	10		Input Point Implementation Level
32	B	sp_OP	10		Output Point Implementation Level
33	B	sp_VC	10		View Display Implementation Level
34	B	sp_PC	10		Positioning System Movement Commands Implementation Level
35	B	sp_TC	10		Tour Control Command Implementation Level
36	B	sp_SC	10		Salvo Control Command Implementation Level
37	B	sp_OC	10		Output Control Command Implementation Level
38	B	sp_EA	10		Activate Event Reporting Implementation Level
39	B	sp_ER	10		Event Report Implementation Level
40	B	sp_EK	10		Event Acknowledge Implementation Level
41	B	sp_ED	10		Deactivate Event Reporting Implementation Level
42	B	sp_ET	10		Event Type Code Implementation Level
43	B	sp_CN	10		Connection Message
44	B	sp_CC	10		Close Connection Message
45	B	cc_DU	10		Define User Message

7.1.3 Close Connection Message – CC (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	cc_reason	1	c	Specifies the reason for closing the connection: • O – Operator Command • S – Shutting Down
2	B	cc_sub_system_id	10	C	The sub-system _id of the subsystem where the login or logout is being performed.

7.1.4 Login / Logout Message – LI (B)

Each sub-system transmits this command to open a session with another sub-system . The command contains a user name and a password. The Q operation code would be used to request login. An A operation code would be the response. The li_status of the response would indicate if the request was granted.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**li_user_name[1,1]**	32	C	Up to 10 characters of user name. User names establish the validity of the user. This is not thought to be an actual station operator at the access control system, rather the system's login to the switching system.
2	B	**li_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where the login or logout is being performed.
3	B	li_user_password	10	C	Up to ten characters of user password. Passwords should not be less than 5 characters.
4	B	li_request	1	'I', 'O'	• I – Request Login On This System • O – Request Logout On This System
5	B	li_status	1	'I', 'O'	• I –User Logged In On This System • O – User Logged Out On This System

7.15 Poll Message – PO (B)

This is a content-less record.

Item	Level	Data Item	LEN	'value' / Format	Description
There is no content to this record type.					

7.2 COMPONENT DATABASE

7.2.1 Views Table – VW (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**vw_view_name [1,1]**	32	C	The unique name of the view.
2	B	**vw_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem for which this view definition applies.
3	S	vw_view_type	1	One of 'S' or 'M'	The type of view. The view may be a Still (S) or Motion (M) view. A still view is defined by the data items in this record. A Motion view is defined in this record through the motion definition name, which is the key into the Motion Definition File Table.
4	B	VW_CAMERA_NAME (CA)	32	C	The unique name of the camera which provides this view unless it is a motion view.
5	B	VW_PRESET_NAME (PR)	10	C	The unique name of the preset which defines the position of the camera and lens to achieve this view if it is required. The field is blank if no preset is required.
6	S	VW_MOTION_DEFINITION_NAME (MD)	10	C	The unique name of the file that contains the motion definition for a motion view if it is a motion view. The field is blank if it is not required.
7	B	VW_VIEW_TITLE_NAME (TL)	10	C	The name of a title that is appropriate to this view. The field is blank if it is not required.

7.2.2 Motion Definition File Table- MD (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**md_motion_definition_name[1,1]**	10	C	The unique name of the motion definition
2	S	**md_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem for where this motion definition file exists.
3	S	MD_MOTION_CAMERA (CA)	10	C	The name of a title that is appropriate to this view. The field is blank if it is not required.
4	S	MD_MOTION_POSITIONING_SYSTEM	10	C	The name of the positioning system.
5	S	md_motion_definition_file_name	10	C	The actual file name of the motion definition file

7.2.3 Preset Definition Table – PR (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**pr_preset_name[1,1]**	10	C	The unique name of the motion definition
2	B	**pr_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this preset definition exists.
3	S	pr_preset_method	1	'N','P'	The method by which this preset is established. N – By preset number, P – By defining a position.
4	B	pr_preset_number	6	C	The preset ID in the specified sub-system .
5	S	pr_positioning_method	1	'A','D'	A – Use right ascension and declination, D – use degrees right, degrees up from max depression
6	S	pr_positioning_origin	1	'H','L', 'N','R'	H – Home is base, L – Max Left is base, N – Use North as base, R – Deflection is relative to current position.
7	S	pr_right_position	6	nnnnnn	Right position or deflection
8	S	pr_vertical_position	6	nnnnnn	Vertical position or deflection
9	S	PR_POSITIONING_SYSTEM_NAME	10	C	The positioning system that implements this preset if required, otherwise blank.
10	S	PR_RECEIVER_NAME	10	C	The receiver that implements this preset if required, otherwise blank.

7.2.4 Camera Table – CA (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**ca_camera_name[1,1]**	32	C	The unique name of a camera.
2	B	**ca_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this camera definition exists.
3	S	ca_camera_type	(4)	C	(Reserved for future – all blank)
4	B	ca_camera_id	10	C	Any id number or other specific identification of the camera.
5	B	CA_RECEIVER_ SUB-SYSTEM _ID(RC)	10	C	The sub-system ID of the receiver referenced above.
6	B	CA_RECEIVER_NAME(RC)	10	C	The unique name of the receiver which controls this camera's position.
7	S	ca_camera_port_id	10	C	The port definition appropriate to locating this camera on the switching system.

7.2.5 Title Table – TL (B)

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**tl_title_name[1,1]**	10	C	The name for the title. Title names are not unique.There is one title for each tl_nls_code. tl_title_name plus tl_nls_code uniquely define a specific set of text..
2	B	**tl_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem for where this title definition record exists.
3	S	tl_title_text	30	C	The text of the title in the language specified by the tl_nls_code.
4	B	**tl_nls_code [1,3]**	4	C	The NLS specification for the view title.
5	S	tl_nls_character_set	12	C	The character code set applicable to display or printing of this title.

7.2.6 Title Display Parameters Table – TP (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**tp_title_display_parm_name[1,1]**	10	C	
2	S	**tp_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem for where this title parameter record exists.
3	S	tp_background_color	11	nnn.nnn.nnn	The eleven characters are divided into three groups of three integers. Each three-integer group provides the color value for a basic color. Valid values for nnn are 000 through 255. From left to right, the nnn groups specify RED, GREEN, BLUE respectively.
4	S	tp_border_color	11	nnn.nnn.nnn	The eleven characters are divided into three groups of three integers. Each three-integer group provides the color value for a basic color. Valid values for nnn are 000 through 255. From left to right, the nnn groups specify RED, GREEN, BLUE respectively.
5	S	tp_font_size	3	nnn	The font size in points.
6	S	tp_font	20	C	The font name
7	S	tp_location_x	4	nnnn	The value of the x coordinate where the left side of the screen has an x value of 0. Only positive x values are allowed.
8		tp_location_y	4	nnnn	The value of the y coordinate where the top side of the screen has an y value of 0. Only positive y values are allowed and increasing y values move down the screen.
9	S	tp_fade_option	4	C	0 = No Fade, 1 = Fade, Other characters specify fade options specific to a manufacturer.
10	S	tp_fade_time	4	nnnn	A fade time interval specified in units of 100ms.
11	S	tp_flash_time	4	nnnn	A flash duration time interval specified in units of 100ms. If the value is 0 there is no flashing. If the value is non-zero, for example 1, the display is bright for the value (e.g. 1) and dim for the value (e.g.1) so that the complete cycle takes two times the value. Bright and dim are of equal durations.

7.2.7 Monitor Table – MN (B)

This table includes traditional monitors plus any other reasonable video destination or display device.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**mn_monitor_name [1,1]**	32	C	The unique name of the monitor.
2	B	**mn_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem for where this motion definition file exists.
3	S	mn_alarm_authorization	1	'Y' or 'N'	May this monitor be associated with a monitor list?
4	S	mn_monitoring_list_type	1	'C' or 'I'	Monitors a common or central list of alarms or an individual 'I' list of alarms.
5	S	MN_MONITORING_LIST_NAME	10	C	If monitoring list type = I, the name or identifier of the list.
6	S	mn_monitoring_list_process	1	C	The algorithm to use to service the list. • A – Auto-Sequential: Alarms in the list are automatically displayed in sequence for a dwell time. The operator must view the alarm and acknowledge it. If acknowledged, it is removed from the list. If not acknowledged, it remains in the list and will be redisplayed in turn. • K – Auto-sequential, Auto-acknowledge • F – FIFO • L - LIFO • P - Priority
7	S	mn_monitoring_sequential_dwell	2	nn	The dwell time in seconds for the display of the view when the monitoring list process = 'A'.
8	S	mn_last_alarm_action	1	C	• P - Restore The View Displayed Prior To Receiving Any Alarms • B - Display A Blank Screen • L - Retain The Last Alarm View • D - Display The Monitor's DEFAULT View
9	S	mn_nls_code	4	C	The default NLS specification for the monitor.
10	S	MN_DEFAULT_VIEW_NAME	32	C	The default view to be displayed at this

Item	Level	Data Item	LEN	'value' / Format	Description
		(VW)			monitor
11	S	MN_DEFAULT_VIEW_TITLE_NAME (TL)	10	C	The title to be associated with this view.
12	S	MN_DEFAULT_VIEW_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of this view.
13	S	MN_VIEW_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of a view.
14	S	MN_ALARM_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of an alarm.
15	S	MN_SECTOR_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of a sector.
16	S	MN_PRESET_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of a preset.
17	S	MN_TIME_DATE_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of time and date.
18	S	MN_MONITOR_TITLE_NAME (TL)	10	C	The title of this monitor.
19	S	MN_MONITOR_DISP_PARM_NAME (TP)	10	C	The display parameters associated with display of this monitor.
20	B	mn_monitor_characters_wide	4	nnnn	The width of the monitor or display in characters.
21	B	mn_monitor_characters_high	4	nnnn	The height of the monitor or display in characters.

7.2.8 Monitor Lists Table – ML (S)

This message is used to create or define a monitor list. A monitor list is a name for a list that alarms may be added to and deleted from for a monitor.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**ml_monitoring_list_name[1,1]**	10	C	Must be the unique name of a list
2	S	**ml_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this monitor list definition exists.
3	S	ml_monitoring_list_type	1	'M' or 'I'	Monitors list is the master list of alarms 'M' or an individual 'I' list of alarms.

7.2.9 Salvos Table – SA (S)

A Salvo is a list where each item in the list includes the specification of a monitor, a view, and a speed expressed as a time interval by which a positioning system should implement the view, given that such movement is necessary. Salvo name, monitor name, and view name together uniquely define a record.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**sa_salvo_name [1,1]**	10	C	The name of the salvo.
2	S	**sa_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem for where this salvo exists.
3	S	**sa_monitor_name (2)**	32	C	The associated monitor.
4	S	**sa_view_name (3)**	32	C	The view name.
5	S	sa_positioning_system_delay	4	nnnn	Time interval expressed in units of 100ms required to establish the correct view on a designated monitor using the slowest positioning system.

7.2.10 Tours - TR (S)

A tour is a collection of views or salvos that are displayed in a prearranged sequence. This record is the master record that defines the tour. The tour step TS record definition follows this definition.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**tr_tour_name [1,1]**	10	C	A unique tour name
2	S	**tr_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem where this tour definition exists.
3	S	tr_tour_type	1	"V" or "S"	V - View or S – Salvo. Tours must be composed of all V or all S steps.
4	S	tr_last_step_action	1	'P', 'B', 'L', 'D', 'N', 'R'	• P - Restore The View Displayed Prior To Beginning The Tour • B - Display A Blank Screen(S) • L - Retain The Last Alarm View(S) • D- Display The Monitor's DEFAULT View(S) • N - Activate Next Tour • R – Repeat Tour for the tr_tour_repeat_limit times
5	S	TR_NEXT_TOUR_ NAME (TR)	10	C	Name of tour to activate upon conclusion of this tour
6	S	tr_tour_repeat_limit	3	nnn	Number of times to cycle this tour. 000 means do not repeat, 999 means continuous

7.2.11 Tour Steps – TS (S)

This record defines a step in a tour.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**ts_tour_name [1,1]**	10	C	The name of the tour which owns this step.
2	S	**ts_sub_system_id [1,2]**	10	C	The sub-system _id of the subsystem where this tour step definition exists.
3	S	**ts_tour_step_number [1,3]**	3	nnn	The number of this step.
4	S	TS_VIEW_SALVO_NAME (SA) OR (VW)	32	C	The name of the view or salvo as defined by tr_tour_type in the TR record to use for this step. The name is left justified, blank filled. If the tr_tour_type is 'S' or salvo, only the leftmost ten characters will be used. If the tr_tour_type is 'V' or view all 32 characters will be used.
5	S	ts_tour_step_type	1	'V', 'S'	V - View or S – Salvo. Tours must be composed of all V or all S steps.
6	S	ts_positioning_system_delay	4	nnnn	Time interval expressed in units of 100ms required to establish the view on using the positioning system. If this is a salvo tour, the salvo positioning delay will apply.
7	S	ts_dwell_time	3	nnn	The time is specified in 100ms increments.

7.2.12 Holidays List Table – HL (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**hl_holiday_list_name[1,2]**	10	C	The unique name of this Holiday day.
2	S	**hl_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this preset definition exists.
3	S	hl_holiday_name	32	C	The name of the Holiday
4	S	hl_holiday_date	8	yyyymmdd	The date of the holiday day.

7.2.13 Schedule Table – SD (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**sd_schedule_name[1,1]**	10	C	Unique name of a schedule
2	S	**sd_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this scheduled definition exists.
3	S	sd_schedule_start_date_time	14	n(14)	yyyymmddhhmmss
4	S	sd_schedule_stop_date_time	14	n(14)	yyyymmddhhmmss
5	S	sd_schedule_day_of_week	7	'bbbbbbb'	Days of week specified by 0 or 1 in seven character field. Position 1 is Sunday. Position 7 is Saturday. 0 indicates not applicable in the schedule, 1 indicates applicable. '0000000' indicates no days applicable. '1111111' indicates all days applicable. '0111110' indicates Monday through Friday applicable.
6	S	sd_schedule_dow_time	6	nnnnnn	hhmmss , 2400 hour clock time
7	S	sd_schedule_type	1	'I', 'S'	The type of schedule: • I – Interval type with beginning and ending time. • S – Specific time using sd_schedule_start_date_time • R – Repeated at sd_schedule_dow_time for days with a '1' in their position in sd_schedule_day_of_week
8		SD_HOLIDAY_LIST_NAME (HL)	10	C	Name of holiday list applicable to this schedule
9		sd_holiday_active_flag	1	b	Flag (1=Yes, 0=No) indicating if this schedule is sensitive to holidays.

7.2.14 Scheduled Event Table – SE (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**se_scheduled_event_name [1,1]**	10	C	The name of this scheduled event.
2	S	**se_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this scheduled definition exists.
3	S	SE_SCHEDULE_NAME (SC)	10	C	Defines the schedule for this event
4	S	SE_RESPONSE_NAME (RD)	10	C	Defines the response to activation of this event.

7.2.15 Event Codes Table – EC (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**ec_event_code [1,1]**	4	C	This is a unique event code
2	S	**ec_sub_system_id[1,2]**	10	C	The source id of the subsystem that can implement this event.
3	S	ec_event_description	32	C	This is a descriptive name of the event code

7.2.16 Response Definition Table – RD (S)

This set of data defines events and how they are to be treated.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**rd_response_name{2}**	10	C	The name of an response.
2	S	RD_RESPONSE	32	C	The description of the local response to be implemented.

7.2.17 Event Response Table – ER (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**er_event_code[1,1]**	4	C	An event code
2	S	**er_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this event response definition exists.
3	S	er_component_name[1,3]	32	C	The name of the reporting component on the sub-system defined by er_ sub-system _id.
4	S	ER_RESPONSE_NAME(RD)	10	C	The name of the response to implement.

7.2.18 User Interface – UI (B)

Item	Level	Data Item	Len	'value' / Format	Description
1	B	**ui_user_login_name[1,1]**	32	C	Unique user interface name
2	B	**ui_sub_system_id[1,2]**	10	C	The sub-system _id of a subsystem where this user is authorized.
3	S	ui_user_interface_priority	2	nn	Priority – The priority level of this user interface relative to others. This is used to manage contention over system resources.
4	S	ui_user_interface_seize_option	1	n	Seize Control Options – The authority of this user interface to seize control over system resources used by or currently controlled by another user interface. Priority levels determine this calculation's results if it is permitted. The highest priority is 1 the lowest is 9, where users with a higher priority may seize control of a resource currently being used by a lower priority user.
5	S	ui_user_interface_default_monitor	32	C	Default Monitor – The default monitor that is used by this user interface.
6	S	ui_default_monitor_ sub-system _id	10	C	The sub-system ID of the above monitor.
7	S	UI_ PRIVELEGE_LEVEL_CODE(PL)	2	cc	A user privilege code from the PL table.
8	B	ui_interface_user_password	10	C	Password for this user on the referenced sub system.
9	S	**ui_user_real_name**	32	C	The name of the user.

7.2.19 Receiver (or Positioning System) Table – RC (B)

Camera Positioning refers to any positioning device of any type. Classically, these systems are controlled by a receiver and have the data specifications of a receiver.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	**rc_receiver_name[1,1]**	10	C	The name or other ID of the positioning system. Also version and type information would be included. The associated camera would be included in this definition. Many systems may use the camera ID to name the positioning system.
2	B	**rc_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this receiver or positioning system exists.
3	S	rc_receiver_type	10	C	Manufacturer specific descriptive information.
4	S	rc_receiver_address_type	1	C	C – Camera Address, R – Receiver Address
5	S	RC_RELATED_CAMERA(CA)	32	C	The name of the associated camera.
6	S	RC_RELATED_CAMERA_ SUB-SYSTEM _ID(CA)	10	C	The sub-system ID of the above camera.
7	S	RC_DEFAULT_POSITION_VIEW(VW)	32	C	The definition of the default position may include more than just a camera position (orientation).
8	S	RC_DEFAULT_POSITION_ SUB-SYSTEM _ID(VW)	10	C	The sub-system ID of the above default view.

7.2.20 Sector Title Table – ST (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**st_receiver_name[1,1]**	10	C	The name or other ID of the receiver for which this record applies.
2	S	**st_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem where this preset definition exists.
3	S	st_sector_position	10	C	Defines by manufacturer the specific sector.
4	S	ST_SECTOR_TITLE_NAME(TL)	10	C	Title name associated with sector.

7.2.21 Input Point Table – IP (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**ip_input_name [1,1]**	10	C	The name of the input point that is unique.
2	S	**ip_sub_system_id [1,2]**	10	C	The source id of the subsystem that owns this input point.
3	S	IP_SOURCE_LOCATION(2)	10	C	The location or address of this device or originating component within the sub-system to which it belongs. For example, this could be a receiver name.
4	S	ip_source_table	2	cc	The table where this source may be found.
5	S	rc_normal_state_aux_inp_1	1	C	The following options are valid for inputs: • NO2 – Normally Open, 2 State • NO4 – Normally Open, 4 State • NC2 – Normally Closed, 2 State • NC4 – Normally Closed, 4 State

7.2.22 Output Point – OP (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**op_output_name [1,1]**	10	C	The name of the output point that is unique.
2	S	**op_sub_system_id [1,2]**	10	C	The source id of the subsystem that owns this output point.
3	S	op_source_location	10	C	The location or address of this device or originating component within the sub-system to which it belongs. For example this could be a receiver.
4	S	op_source_table	2	cc	The table where this source may be found.
5	S	rc_default_position_aux_out_1	1	C	The following options are valid for outputs: • NO – Normally Open • NC – Normally Closed

7.2.23 Privilege Level Table – PL (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**pl_privelage_level_code [1,1]**	2	cc	Code for user privilege level
2	S	**pl_sub_system_id [1,2]**	10	C	The source id of the subsystem where this user exists.
3	S	pl_privilege	32	C	As implemented in the referenced system.

7.2.24 Configure Logging Table – CL (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	**cl_event_code[1,1]**	4	C	This data item would specify an event type. See Event Code Table.
2	S	**cl_sub_system_id[1,2]**	10	C	The sub-system _id of the subsystem for which this logging definition applies.
3	S	cl_logging	1	C	This option would specify the logging option to apply to logging instances of this event: • E – Enable logging at all times • D – Disable logging at all times • S – Log during active intervals of the log schedule in this record. • X – Log during inactive intervals of the log schedule in this record.
4	S	CL_LOG_SCHEDULE(SD)	10	C	A schedule name from the schedule file.
5	S	CL_COMPONENT_NAME	10	C	The name of a component which originates events of this type. Specification of a component would cause the system to apply this rule to an occurrence of this event at this component, overriding a general (non-device-specific) rule.
6	S	CL_COMPONENT_ SUB-SYSTEM _ID	10	C	sub-system ID of the above referenced component.
7	S	cl_component_table	2	cc	The table or file where this device may be found. This might be VW, MN, IP, etc. The valid entries are those components which produce transactions.

7.3 COMMAND MESSAGES

7.3.1 Show View Command – VC (B)

This refers to issuing a command to a switching system telling it to display a specific view on a specific monitor.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	VC_VIEW_NAME(VW)	32	C	The name of the view on the sub-system to which the command is sent.
2	S	VC_VIEW_SUB_SYSTEM _ID	10	C	The sub-system id of the where this view exists.
3	B	VC_MONITOR_NAME(MN)	32	C	The name of the monitor on the sub-system to which the command is sent.
4	S	VC_MONITOR_SUB_SYSTEM _ID	10	C	The sub-system id of the where this monitor exists.

7.3.2 Movement Command - MC (B)

This refers to the movement commands issued by a sub-system through the switching systems to a camera positioning system.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	MC_RECEIVER_NAME(RC)	10	C	The name of the receiver or positioning system on the sub-system to which this command is directed.
2	S	MC_RECEIVER_SUB_SYSTEM _ID	10	C	The sub-system id of the where this receiver exists.
3	B	mc_movement_operation	1	c	• S – Stop All Functions • C – Change
4	B	mc_movement_pan	1	c	• L – Pan Left • R – Pan Right
5	B	mc_movement_tilt	1	c	• U – Tilt Up • D – Tilt Down
6	B	mc_movement_zoom	1	c	• T – Telephoto • W - Wide
7	B	mc_movement_focus	1	c	• N – Focus Near • F – Focus Far
8	B	mc_movement_iris	1	c	• O – Iris Open • C – Iris Close
9	B	mc_movement_speed_pan	7	nnn.nnn	The speed in degrees per second. Allowed values are 0 to 999 degrees per second where the sub-system to which this command is directed would implement the requested speed to the best of its ability.
10	B	mc_movement_speed_tilt	7	nnn.nnn	The speed in degrees per second. Allowed values are 0 to 999 degrees per second where the sub-system to which this command is directed would implement the requested speed to the best of its ability.
11	B	mc_movement_time_out	4	nnnn	A time out in 100ms increments.

7.3.3 Tour Control Command - TC (S)

This refers to the commands to the switching systems to activate a specific tour at a specific monitor.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	TC_TOUR_NAME	10	C	The name of the tour
2	S	TC_TOUR_SUB_SYSTEM_ID	10	C	The sub-system id of the where this tour exists.
3	S	tc_tour_command	1	c	Allowed values: • B – Begin • E - End
4	S	tc_command_duration	10	nnnnnnnnnn	An associated duration in seconds
5	S	tc_tour_monitor	10	c	Related Monitor for the command

7.3.4 Salvo Control Command – SC (S)

This refers to the commands to the switching systems to activate a specific salvo.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	SC_SALVO_NAME	10	C	Name of Salvo to be activated on the sub-system to which this command has been directed.
2	S	SC_SALVO_SUB_SYSTEM _ID	10	C	The sub-system id of the where this salvo exists.

7.3.5 Output Control Command – OC (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	OC_OUTPUT_NAME	10	C	The name of the output to be operated on the sub-system to which this command has been directed.
2	S	OC_OUTPUT_SUB_SYSTEM_ID	10	C	The sub-system id of the where this output exists.
3	S	oc_output_command	2	cc	Allowed Values: • OP – Open • CL – Closed
4	S	oc_duration	7	nnnnnnn	Duration in 100 ms increments that the output command is to be implemented. If the value is zero the duration is permanent.

7.3.6 Portal Control Command – PC (F)

Item	Level	Data Item	LEN	'value' / Format	Description
1	F	PC_PORTAL_NAME	20	C	The name of the portal to which this command shall be applied on the sub-system to which this command is being sent
2	S	PC_POTRAL_SUB_SYSTEM_ID	10	C	The sub-system id of the where this portal exists.
3	F	pc_portal_command	2	cc	Allowed Values: • UL – Unlock • LK- Lock • GA- Grant Access
4	F	pc_duration	7	nnnnnnn	Duration in 100 ms increments that the output command is to be implemented. If the value is zero the duration is permanent.

7.3.7 Badge Access Command – BA (F)

To be addressed in the next revision

7.3.8 Activate Event Reporting – AR (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	ar_user_name	10	C	The user's log-in name
2	S	ar_user_password	10	C	The user's password

7.3.9 Save Video Record Command – SV (S)

This refers to the commands to the switching systems to activate a specific salvo.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	SV_VIEW_NAME	10	C	Name of view on the sub-system to which this command to be saved.
2	S	SV_VIEW_SUB_SYSTEM_ID	10	C	The sub-system id where this view exists.
3	S	sv_event_date_time	14	n(14)	yyyymmddhhmmss
4	S	sv_seconds_before	4	nnnn	The number of seconds before the event that should be a part of the saved image.
5	S	sv_seconds_after	4	nnnn	The number of seconds after the event that should be a part of the saved image.
6	S	sv_save_type	1	C	"V" - VTR, F – Data File
7	S	sv_save_path	128	C	The path to the location where the file should be saved.
8	S	sv_save_file_name	32	C	The name of the save file.
9	S	sv_file_type	3	C	One of : "jpg", "mpg" mpg refers to MPEG II. Others will be added in the future.
10	S	SV_MONITOR_NAME	32	C	The name of the monitor on which this file should be displayed.
11	S	SV_MONITOR_SUP_SYSTEM_ID	10	C	The name of the sub-system on which this monitor is found.
12	S	sv_monitor_speed	1	C	"T" – Time Lapse, "R": - Real time. The actual speed is determined by the settings on the recording device.

7.3.10 Retrieve Video Record Command – RV (S)

This refers to the commands to the switching systems to activate a specific salvo.

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	RV_CAMERA_NAME	10	C	Name of camera on the sub-system to which this command has been directed.
2	S	RV_CAMERA_SUB_SYSTEM_ID	10	C	The sub-system id where this camera exists.
3	S	rv_source_type	1	C	"V"- VTR, "F" – Data File
4	S	rv_event_date_time	14	n(14)	yyyymmddhhmmss
5	S	rv_seconds_before	4	nnnn	The number of seconds before the event that should be a part of the saved image.
6	S	rv_seconds_after	4	nnnn	The number of seconds after the event that should be a part of the saved image.
8	S	rv_source_path	128	C	The path to the location where the file should be saved.
9	S	rv_source_file_name	32	C	The name of the save file.
10	S	rv_source_file_type	3	C	One of : "jpg", "mpg"
11	S	rv_save_type	1	C	"V"- VTR, "F" – Data File
12	S	rv_save_path	128	C	The path to the location where the file should be saved.
13	S	rv_save_file_name	32	C	The name of the save file.
14	S	rv_save_file_type	3	C	One of : "jpg", "mpg" mpg refers to MPEG II. Others will be added in the future.
15	S	RV_MONITOR_NAME	32	C	The name of the monitor on which this file should be displayed.
16	S	RV_MONITOR_SUB_SYSTEM_ID	10	C	The name of the sub-system on which this monitor is found.
17	S	rv_monitor_speed	1	C	"T" – Time Lapse, "R": - Real time. The actual speed is determined by the settings on the recording device.

7.3.11 Event Acknowledge – EA (F)

Description forthcoming in the next revision.

7.3.12 Cancel Event Reporting – XR (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	XR_USER_NAME	10	C	The user's log-in name
2	S	XR_USER_SUB_SYSTEM_ID	10	C	The sub-system id of the where this user exists.
3	S	xr_user_password	10	C	The user's password

7.3.13 Time & Date Message – TD (B)

Time and Date controls include the format of the items. Time values should be specified using hh for hours, mm for minutes, and ss for seconds. Date values should be specified using YYYY for year, MM for month, and DD for day. Day of week should be specified by the letter d where Sunday is 1 and Saturday is 7. A format specification specifies the display format of the date and time and the day of the week. These are language specific.

Item	Level	Data Item	LEN	'value' / Format	Description
1	B	td_new_time	6	nnnnnn	hhmmss, New time in 24 hour format.
2	B	td_new_date	8	YYYYMMDD	New date.
3	B	td_new_dow	1	d	New day of week. Valid values are "1" through "7" where Sunday is "1" and Saturday is "7".
4	B	td_got_name	10	C	My time sub-system 's sub-system _id
5	B	td_time_zone	2	nn	My time zone from 0 to 23 from International Date Line
6	B	td_dlst_value	3	nnn	Current minutes ahead or behind I am relative to standard time for this time zone.
7	B	td_st_direction	1	b	1 – Ahead, 0 – Behind, cs_dlst_value is ahead or behind local standard time.

7.4 EVENT MESSAGES

7.4.1 Event Reporting Message – EM (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	em_event_name	10	C	The name of the event that is being reported.
2	S	em_event_sub_system_id	10	C	The sub-system id of the where this event occurred.
3	S	em_event_date	8	yyyymmdd	The date of the event.
4	S	em_event_time	6	nnnnnn	hhmmss, The time of the event on a 24-hour clock.
5	S	em_event_time_zone	2	nn	The time zone relative to Greenwich Mean Time (GMT) of the reported date and time.
6	S	em_event_status	1	C	A code representing a contextually valid statement of the event's status. • A – Acknowledged • N – New • R – Released
7	S	em_event_id	10	nnnnnnnnnn	A unique identifier of this instance of the event.
8	S	em_response_component	32	C	A component name which is required for this event. The component table where this component is to be found is defined by the corresponding component type specification in the event code table defined by fields 2 and 3.
9	S	em_response_table	2	CC	The component table type of the above component.
10	S	em_event_code	4	C	The event code that describes the reported activity.

7.4.2 Logging Message – LM (S)

Item	Level	Data Item	LEN	'value' / Format	Description
1	S	lm_event_code	4	C	The event code which describes the event being logged.
2	S	lm_event_sub_system_id	10	C	The sub-system id of the where this event occurred..
3	S	lm_component_name	32	C	The component name originating the event.
4	S	lm_component_ sub-system _id	10	C	The source id of the subsystem that defines lm_component_name.
5	S	lm_component_table	2	cc	The table type of component originating the event; that is, where the name may be looked up.
6	S	lm_event_date	8	yyyymmdd	The date of the event.
7	S	lm_event_time	6	nnnnnn	hhmmss, The time of the event on a 24-hour clock.
8	S	lm_event_time_zone	2	nn	The time zone relative to Greenwich Mean Time (GMT) of the reported date and time.
9	S	lm_event_status	1	c	A code representing a contextually valid statement of the event's status. • A – Acknowledged • N – New • R – Released • S - Reset
10	S	lm_event_id	10	nnnnnnnnnn	A unique identifier of the event.

7.4.3 Analog Video - (F)

7.4.4 Digital Video - (F)

8. MESSAGE EXCHANGE & MANAGEMENT

This section illustrates the use of the message structures defined above to establish, cancel and otherwise manage sessions between sub-systems; to report, update, delete, query, or receive updates, deletes, reports, or queries from connected sub-systems about data; to receive or transmit reports of activity to connected sub-systems; and to receive or transmit commands to a connected sub-system . Each illustration presents a tabular description of the transaction and an associated schematic representation of the message exchange.

Table 4 defines the meaning of Operation Codes (S,Q,U,D,A). Table 5 provides a list of all Record Type Codes (RTC). Table 6 defines Return Codes and Table 6.1 defines currently approved Reason Codes.

The schematic reports the values of important data items by specifying the system, the data item, and the value in the format:

(system): (data item name) = (value)

Example: B:cs_value=U

The allowed values for system are A, B, and M. A and B denote systems using the interface protocol. M denotes a value in a message set by the transmitting system that is specific to an instance of a message and not generally related to a value to be retained in the transmitting systems local database.

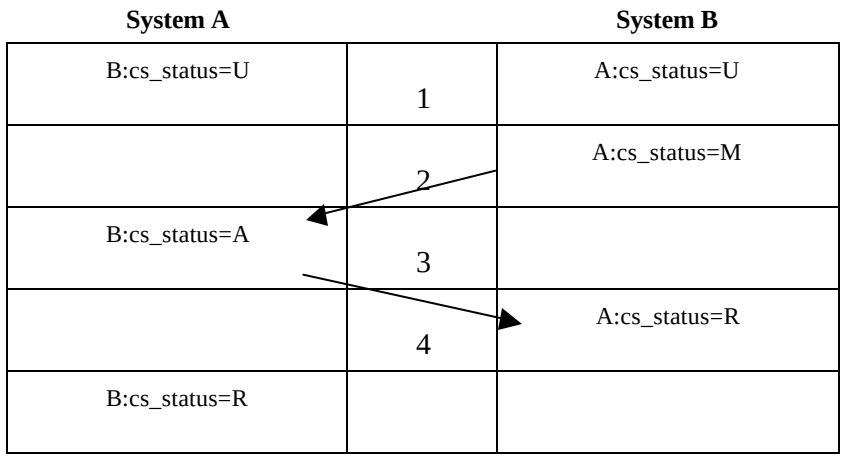
8.1 CONNECTION MANAGEMENT

This section describes the process of making connections, breaking connections, timeout, recovery of a connection, login, logout, and polling.

8.1.1 Making a Connection

System A and System B are assumed to have knowledge of each other or to be allowed to accept definition of a connection from a connecting system using a CS message. If neither of these case is true, then there will be no response to this message. Both systems are assumed to have the state of the other system as U.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	CS	N/A	The CS record is transmitted as shown with the value of cs_status = U.
2	S	CS	N/A	The CS record is transmitted as shown with the value of cs_status = M.
3	S	CS	N/A	The CS record is transmitted as shown with the value of cs_status = A.
4	S	CS	N/A	The CS record is transmitted as shown with the value of cs_status = R.
				Both Systems now report the status of the other system as R.

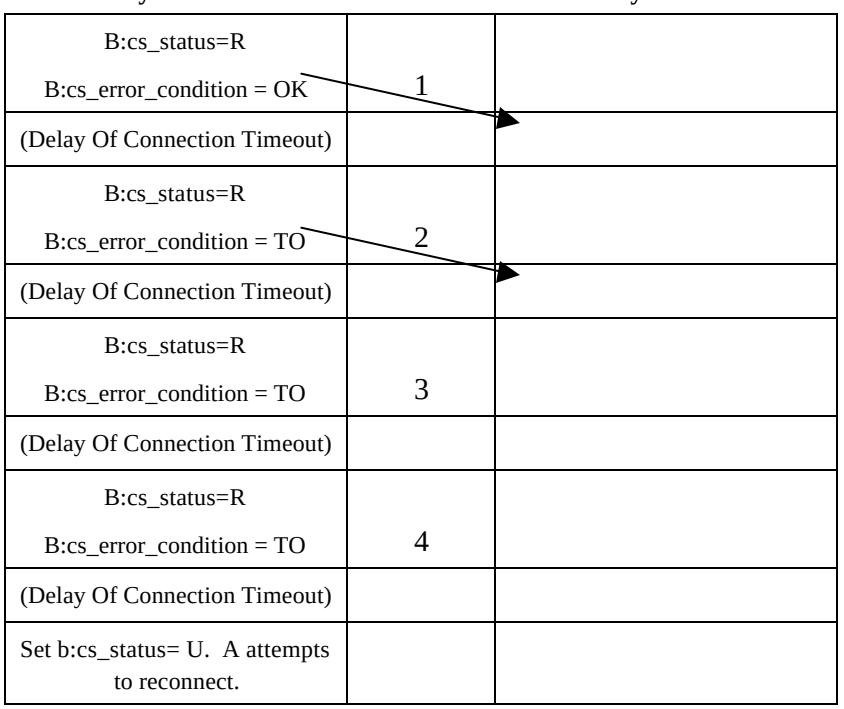


Note: After Message 3, System B sets the status of System to R. This follows since receipt of an A message from System A confirms A's receipt of a message (Message 2) and the positive processing of that message.

8.1.2 Message Timeout

System A and System B are assumed to a valid connection with status R. This applies to any message requiring response.

##	OC	RTC	Return Code	Remark / Key Data Content
1	**	**	N/A	Initial Attempt
2	**	**	N/A	Retry =1
3	**	**	N/A	Retry = 2
4	**	**	N/A	Retry = 3



Note: Reporting of the lost connection is not specified here. The system detecting the outage should begin to reestablish the connection immediately.

8.1.3 Close Connection

System A and System B are assumed to a valid connection with status R.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	CS	N/A	cc_error_condition should explain the reason for the close of the connection. Typical values could be CL, RL, and NS. (See CS message definition)

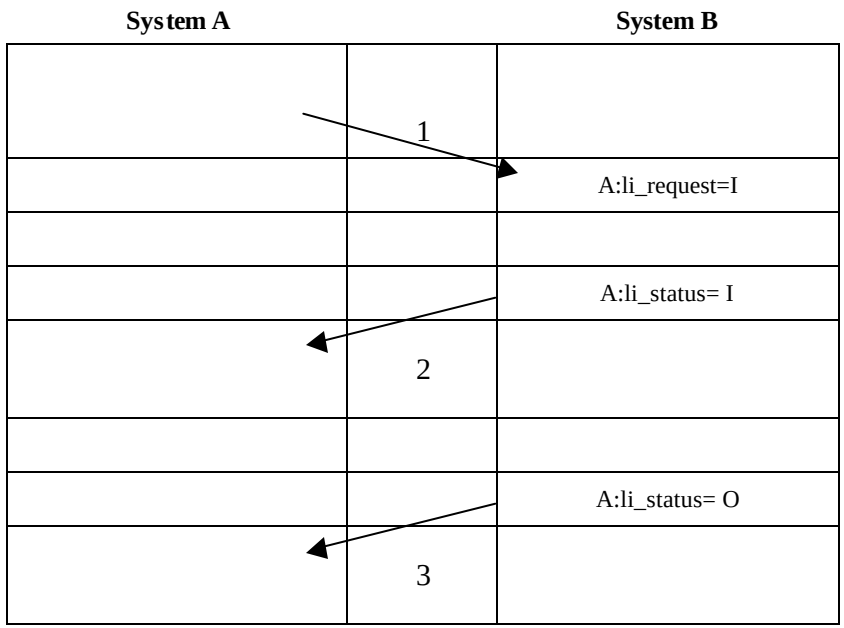
System A		System B
B:cs_status=R B:cs_error_condition = OK	1 →	

Note: Reporting of the lost connection is not specified here. Efforts to reestablish a connection are not specified here.

8.1.4 Login (With cs_status=R)

System A and System B are assumed to a valid connection with status R. Two cases are presented, login approved and denied. Login is asymmetric, i.e., A may be logged in to B without B being logged into A.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	li	N/A	System A establishes M:Li_request=I from A:li_request=1 and A:li_status=O and requests login approval by B
	S	li	OK	Case: Login Approved
2				A:li_request=I A:li_status = I
	S	li	RJ	Case: Login Denied
3				A:li_request=I A:li_status = O

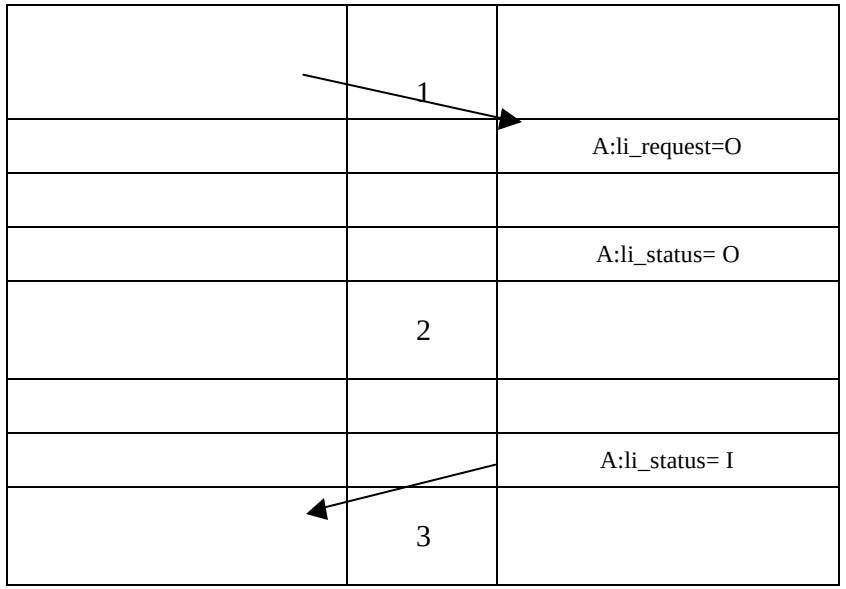


Note: When a system determines that a particular login is required is not specified here.

8.1.5 Logout (With cs_status=R)

System A and System B are assumed to a valid connection with status R. Two cases are presented, logout approved and denied. Login is asymmetric, i.e., A may be logged in to B without B being logged into A.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	li	N/A	System A establishes M:Li_request=O from A:li_request=O and A:li_status=I and requests logout approval by B
				Case: Logout Approved
2		li	OK	A:li_request=O A:li_status = O
				Case: Logout Denied
3		li	RJ	A:li_request=O A:li_status = I



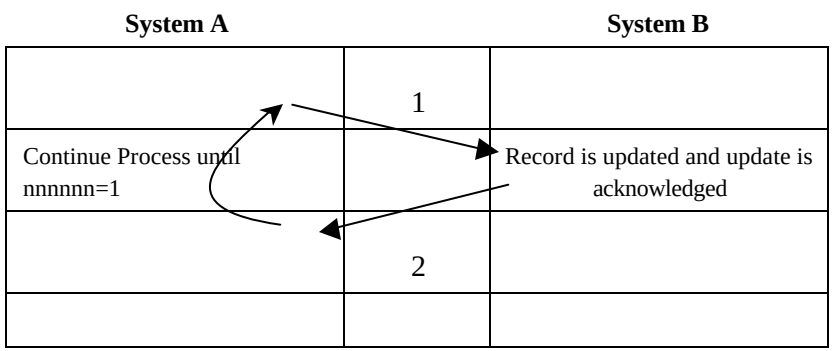
Note: When a system determines that a particular logout is required is not specified here nor is its action to a rejection.

8.2 DATABASE UPDATES

8.2.1 Database Updates (Successful)

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update.

##	OC	RTC	Return Code	Remark / Key Data Content
1	U	**	nnnnnn	The value of nnnnnn is the count of remaining updates including this update.
2	A	PO	mmmmmm	The value of mmmmmm specifies the update record being acknowledged.



Note: When a system determines that a particular logout is required is not specified here.

8.2.2 Database Updates (with Message Timeout)

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update. Example assumes 3-retry rule.

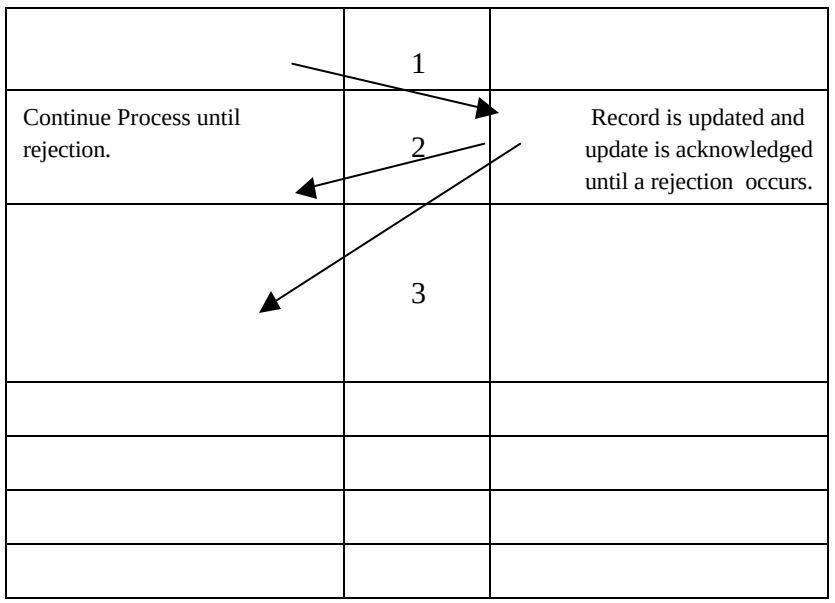
##	OC	RTC	Return Code	Remark / Key Data Content
1	U	**	nnnnnn	The value of nnnnnn is the count of remaining updates including this update.
2	A	PO	mmmmmm	The value of mmmmmm specifies the update record being acknowledged.
				Resend the same record that timed out
				Resend the same record that timed out
				Resend the same record that timed out
				Set cs_status to U, cs_error_condition to TO, Start connection process.

System A		System B
	1	
Continue Process until timeout occurs.		Record is updated and update is acknowledged
	2	
TIMEOUT		
RETRY 1		
TIMEOUT		
RETRY 2		
TIMEOUT		
RETRY 3		
TIMEOUT		

8.2.3 Database Updates (Rejected Cases)

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update. Neither RJ or NO constitute a close connection issue. They are simply failed or rejected transactions.

##	OC	RTC	Return Code	Remark / Key Data Content
1	U	**	nnnnnn	The value of nnnnnn is the count of remaining updates including this update.
2	A	PO	mmmmmm	The value of mmmmmm specifies the update record being acknowledged.
3	A	PO	RJxxxx or NOxxxx	A rejection or No message is received. NO indicated a problem in processing. RJ indicates that the message and its content was OK but for another reason it was rejected. In all cases the reason code xxxx is important.

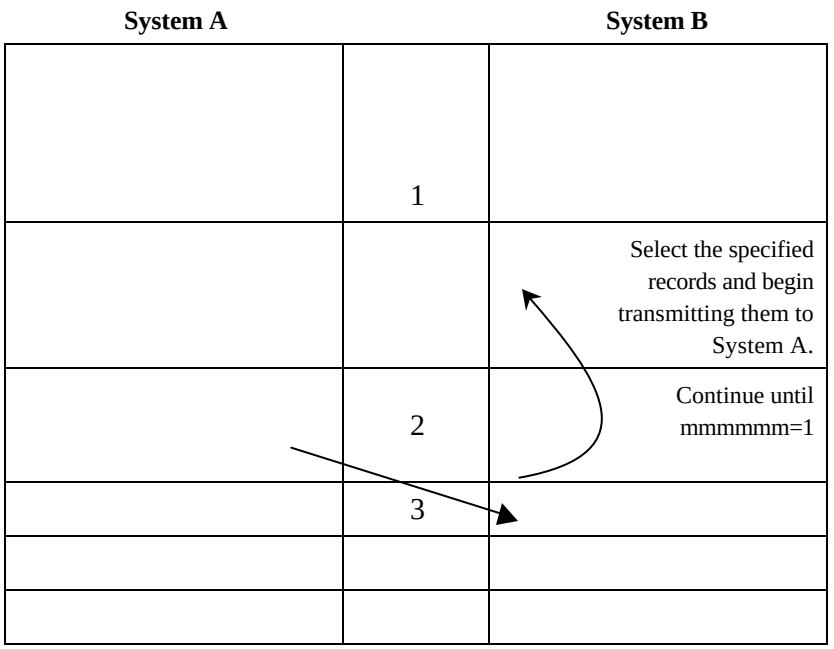


Note: After rejection of any type, the transmitting system must evaluate the reason code and determine the proper action. This specification does not establish the correct action.

8.2.4 Database Query

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update.

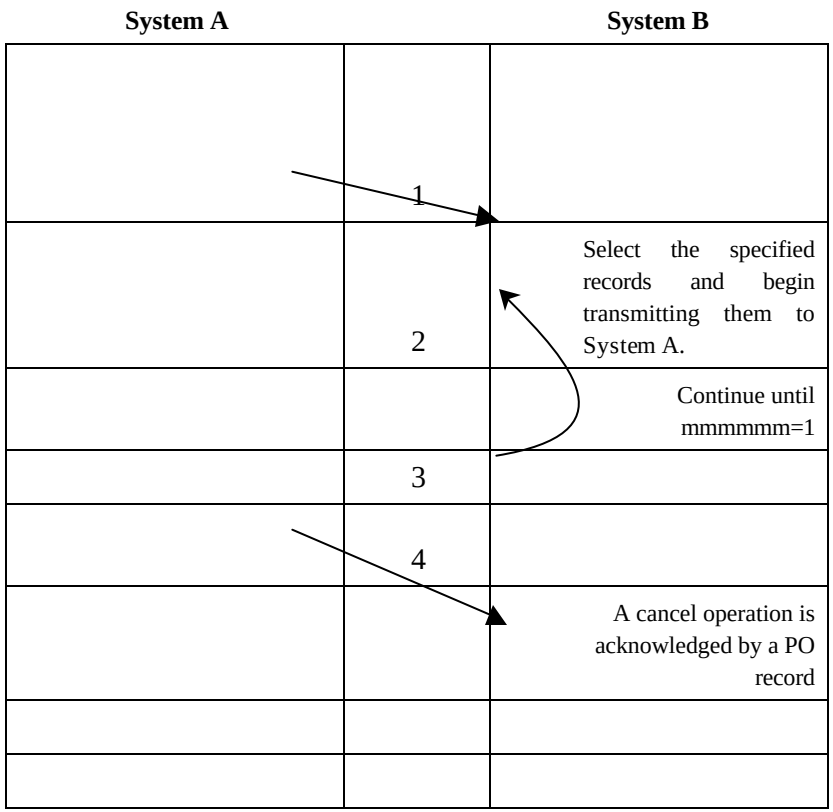
##	OC	RTC	Return Code	Remark / Key Data Content
1	Q	**	nnnnnn	The record data item content will be limited to record specification values or spaces. All records would be selected if a blank record was sent. Nnnnnn will be blank at start and then repeat the last value of mmmmmm.
2	U	**	mmmmmm	The value of mmmmmm specifies the query return records left to transmit including this record
3	A	**	mmmmmm	This acknowledges receipt of record mmmmmm



8.2.5 Cancel Current Operation

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update.

##	OC	RTC	Return Code	Remark / Key Data Content
1	Q	**	nnnnnn	The record data item content will be limited to record specification values or spaces. All records would be selected if a blank record was sent. Nnnnnn will be blank at start and then repeat the last value of mmmmmm.
2	U	**	mmmmmm	The value of mmmmmm specifies the query return records left to transmit including this record
3	A	PO	mmmmmm	This acknowledges receipt of record mmmmmm
4	A	PO	RJcncl	Respond with a return code of RJcncl to cancel any current operation
5	A	PO	OK	



8.2.6 Delete Records

System A and System B are assumed to a valid connection with status R. This process is to be used for any Component Database record update.

##	OC	RTC	Return Code	Remark / Key Data Content
1	D	**	nnnnnn	The record data item content will be limited to record specification values or spaces. All records would be selected if a blank record was sent. Nnnnnn will be blank at .
2	A	PO	Noxxxx, RJxxxx nnnnnn	No and RJ are as previously described. A numerical nnnnnn value indicates the number of records deleted.

System A		System B
	1	
		Select the specified records and delete them
	2	

8.3 COMMANDS

8.3.1 Command Message

System A and System B are assumed to a valid connection with status R. This process is to be used for any Command Message.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	**	NA	
2	A	PO	OK, Noxxxx, RJxxxx	OK, No and RJ are as previously described.

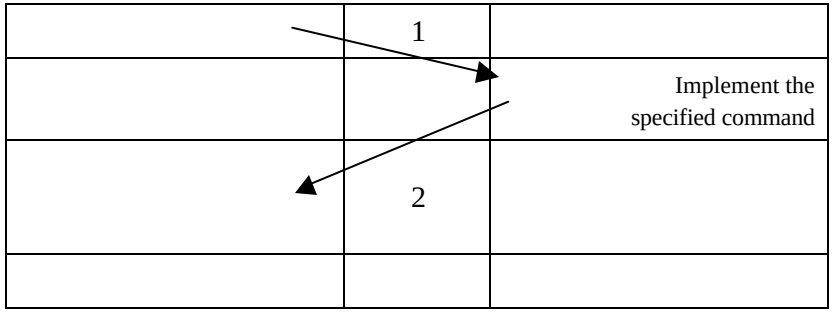
System A		System B
	1	
		Implement the specified command
	2	

8.4 EVENT REPORTS

8.4.1 Event Report

System A and System B are assumed to a valid connection with status R. This process is to be used for any Event Reporting Message.

##	OC	RTC	Return Code	Remark / Key Data Content
1	S	**	NA	
2	A	PO	Noxxxx, RJxxxx nnnnnn	No and RJ are as previously described. A numerical nnnnnn value indicates the number of records deleted.



9. IMPLEMENTATION AND COMPLIANCE TESTING

It is anticipated that an independent third party, such as Underwriter Laboratories, will undertake the task of developing a compliance test. Testing and certification of any product desiring a label for compliance to the SIA TVAC-01 standard would then undergo testing at the third party laboratory.

APPENDIX A: CHECKSUM CALCULATION

CRC-16 (Cyclical Redundancy Check)

The generator polynomial for CRC-16 is $X^{16}+X^{15}+X^2+X^0$.

For the purposes of the SIA protocol, the CRC will be calculated from "seeded" registers. The nature of CRC calculation creates a problem when the CRC register is empty (all 0's) and the message to be sent is sending null characters (all 0's). When this happens, the null bytes being transmitted have no effect on the CRC being generated.

The CRC detects[1]:

Single-bit errors	100%
Double-bit errors	100%
Odd-numbered errors	100%
Burst errors shorter than 16 bits	100%
Burst errors of exactly 17 bits	99.9969%
All other burst errors	99.9984%

A CRC is, fundamentally, the remainder of a polynomial division.

The algorithm to implement a CRC calculation is as follows:

1. Fill the CRC checksum value with the "seed" value. In our case, the seed value is 0xFFFF.
2. Take the high order bit (MSB) of the first byte (it is order sensitive) of the message and left shift the bit into the CRC checksum. Take note of the bit being shifted left off of the checksum.
3. If that bit is a "1", perform an XOR of the CRC checksum and the generator polynomial value.

[1] *"C Programmer's Guide to Serial Communications"* by Joe Campbell, Howard W. Sams & Company

Here is a diagrammatic representation of the algorithm in operation.

HB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	**OPERATION**
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1									Initialize check sum.
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	1	Get message byte.
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		0	1	0	1	0	1	0	1	Shift check sum left one bit.
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	1		Shift message byte left one bit.
1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1		High bit is on (HB), XOR with polynomial.
0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1		1	0	1	0	1	0	1		Shift check sum left one bit.
0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1			Shift message byte left one bit.
0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1			High bit is off (HB).
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1		0	1	0	1	0	1			Shift check sum left one bit.
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1				Shift message byte left one bit.
1	0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1				High bit is on (HB), XOR with polynomial.
0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1		1	0	1	0	1				Shift check sum left one bit.
0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1					Shift message byte left one bit.
0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1					High bit is off (HB).
1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1					Shift check sum left one bit.
1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1		0	1	0	1					Shift message byte

																								left one bit.
1	0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0	1					High bit is on (HB), XOR with polynomial.
0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1		1	0	1					Shift check sum left one bit.
0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0	1						Shift message byte left one bit.
0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0	1						High bit is on (HB) off.
1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1		0	1						Shift check sum left one bit.
1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0	1							Shift message byte left one bit.
1	0	1	1	1	1	1	1	0	1	0	1	0	1	0	1	1	1							High bit is on (HB), XOR with polynomial.
0	1	1	1	1	1	1	0	1	0	1	0	1	0	1	1		1							Shift check sum left one bit.
0	1	1	1	1	1	1	0	1	0	1	0	1	0	1	1	1								Shift message byte left one bit.
0	1	1	1	1	1	1	0	1	0	1	0	1	0	1	1	1								High bit is off (HB).

So, the CRC-16 check sum for the single character message “U” is 0xFD57.

The following is a simple Visual Basic program that will calculate a CRC. The message was input in a text box called “Text1”, this routine was called by the click event on the button “Command1”, and the result was displayed in a text box called “Text2”.

```
Private Sub Command1_Click()

    poly = 2 ^ 15 + 2 ^ 2 + 2 ^ 0

    checksum = 65535                                    ' hex FFFF

    For i = 1 To Len(Text1)                             ' for each byte
    msg = Asc(Mid$(Text1, i, 1))                        ' get then next byte
        For j = 1 To 8                                  ' for each bit in the byte
            msg = msg * 2                               ' shift left one bit
            If msg > 255 Then ext = 1 Else ext = 0      ' remember, was the high bit on?
            msg = msg Mod 256                           ' dump the high bit
            checksum = checksum * 2                     ' shift the checksum left one bit
            If ext = 1 Then checksum = checksum + 1     ' if the msb of the message byte was on,
                                                        ' turn on the low bit of the checksum
            If checksum > 65535 Then                    ' if the high bit of the checksum was on
                checksum = checksum Mod 65536           ' return the checksum to 16 bits and
                checksum = checksum Xor poly            ' xor with the polynomial
            End If
        Next j
    Next i

    Text2 = Hex(checksum)                               ' all done, show the result

End Sub
```

635 Slaters Lane, Suite 110
Alexandria, Virginia 22314

Phone: (703) 683-2075
Fax: (703) 683-2469

E-mail: Standards@SIAOnline.org